

日立マイクロコンピュータシステム

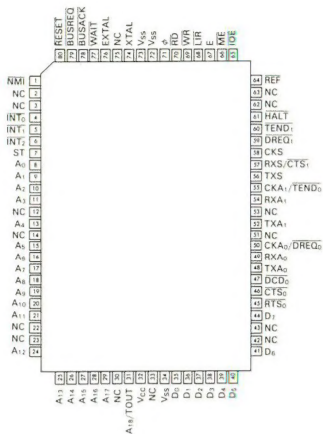
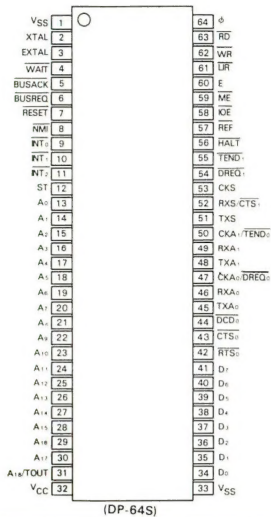
インストラクションポケットブック

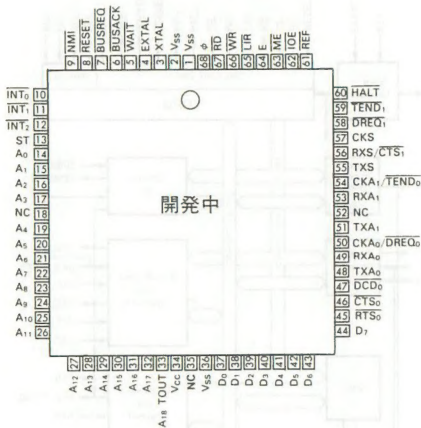
JUN.1986



HITACHI

# ■ ピン配置図 (上面図)



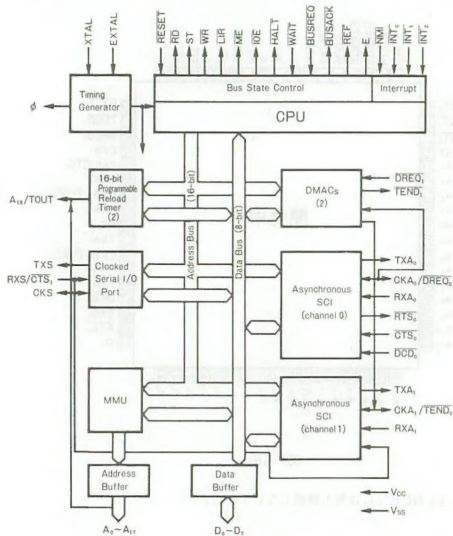


開発中

(CP-68)

注) NCピンには何も接続しないでください。

■ブロック図



# CPUレジスタ

汎用レジスタ

Accumulator A	Flag F
B Register	C Register
D Register	E Register
H Register	L Register

← レジスタセット GR →

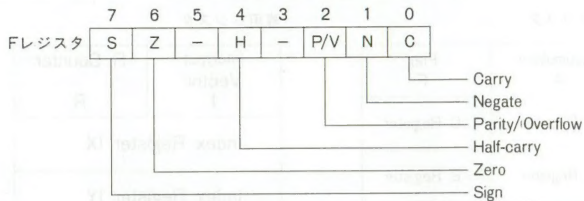
Accumulator A'	Flag F'
B' Register	C' Register
D' Register	E' Register
H' Register	L' Register

← レジスタセット GR' →

専用レジスタ

Interrupt Vector I	R Counter R
Index Register IX	
Index Register IY	
Stack Pointer SP	
Program Counter PC	

## ステータスフラグ



## 割り込み優先順位

割 込 み 要 因	優先度	I L			固 定 コ ー ド				
		b <sub>7</sub>	b <sub>6</sub>	b <sub>5</sub>	b <sub>4</sub>	b <sub>3</sub>	b <sub>2</sub>	b <sub>1</sub>	b <sub>0</sub>
INT <sub>1</sub>	高	*	*	*	0	0	0	0	0
INT <sub>2</sub>	↑	*	*	*	0	0	0	1	0
タイマチャンネル0		*	*	*	0	0	1	0	0
タイマチャンネル1		*	*	*	0	0	1	1	0
DMAチャンネル0		*	*	*	0	1	0	0	0
DMAチャンネル1		*	*	*	0	1	0	1	0
シリアルI/Oポート		*	*	*	0	1	1	0	0
非同期SCIチャンネル0	↓	*	*	*	0	1	1	1	0
非同期SCIチャンネル1	低	*	*	*	1	0	0	0	0

# 各種要求入力の各動作モードにおける受付可否一覧表

現在の Action 要求	現在の 状態 (CPUモード) (I/OSTOPモード)	通常状態 (CPUモード) (I/OSTOPモード)	Waitステート	Refresh サイクル	割り込み アクトレヅジ サイクル	DMA サイクル	BUS RELEASE モード	SLEEP モード	SYSTEM STOP モード
WAIT	受付ける	受付ける	受付けない	受付ける	受付ける	受付けない	受付けない	受付けない	受付けない
Refresh 要求(内蔵リフレッシュ回路によるリフレッシュサイクル挿入要求)	マシンサイクルの切れ目でリフレッシュサイクル挿入	受付けない	受付けない	マシンサイクルの切れ目でリフレッシュサイクル挿入	マシンサイクルの切れ目でリフレッシュサイクル挿入	受付けない	受付けない	受付けない	受付けない
DREQ <sub>0</sub> DREQ <sub>1</sub>	マシンサイクルの切れ目で DMA サイクル挿入	受付けるがマシンサイクルの切れ目まで DMA サイクルは挿入されない	受付ける (※) Refresh サイクル終了後、1 マシンサイクル実行した後に DMA サイクルが挿入される。	受付ける マシンサイクルの切れ目で DMA サイクルが挿入される	受付ける	受付ける (※) バス解放サイクル終了後 1 マシンサイクル実行した後に DMA サイクルが挿入される。	受付けない	受付けない	受付けない
BUSREQ	マシンサイクルの切れ目で BUS RELEASE モードになる	受付けない	受付けない	マシンサイクルの切れ目で BUS RELEASE モードになる	マシンサイクルの切れ目で BUS RELEASE モードになる	BUS RELEASE モードを継続する	受付ける	受付ける	受付ける

(※)

DREQ<sub>0</sub>、DREQ<sub>1</sub>がレベル検出に設定されている場合は受付けられない。

(※※)

INT<sub>0</sub>をモード 0 で使用した場合には受付けられる。バスにのせられた命令の実行終了後NMIのアクトレジサイクルに入る。

現在の状態 Action 要求		通常状態 (CPUモード) (IOSTOPモード)	Waitステート	Refresh サイクル	割込み アクノレッジ サイクル	DMA サイクル	BUS RELEASE モード	SLEEP モード	SYSTEM STOP モード
割   込   み	INT <sub>0</sub> INT <sub>1</sub> INT <sub>2</sub>	命令の最後のマシンスイクルで受付ける	命令の最後のマシンスイクルで受付ける	受付けられない	受付けられない	受付けられない	受付けられない	受付けられて SLEEP モードから通常状態に復帰する	受付けられて SYSTEM STOP モードから通常状態に復帰する
	内蔵 I/O による 割込み 要求	命令の最後のマシンスイクルで受付ける	命令の最後のマシンスイクルで受付ける	受付けられない	受付けられない	受付けられない	受付けられない	受付けられて SLEEP モードから通常状態に復帰する	受付けられない 割込み 要求は発生しない
	NMI	命令の最後のマシンスイクルで受付ける	命令の最後のマシンスイクルで受付ける	受付けられない	受付けられない (※※) アクノレッジサイクル終了後次の命令の最後のマシンスイクルで受付けられる	受付ける DMA は中断される	受付けられない	受付けられて SLEEP モードから通常状態に復帰する	受付けられて SYSTEM STOP モードから通常状態に復帰する

(※) DREQ<sub>0</sub>, DREQ<sub>1</sub> がレベル検出に設定されている場合は受付けられない。

(※※) INT<sub>0</sub> をモード 0 で使用した場合には受付けられる。バスにのせられた命令の実行終了後 NMI のアクノレッジサイクルに入る。



## 各種要求の優先順位

CPUに対する各種のRequest入力は、次の3つに分類されます。

- ① ステート単位で受付け、実行されるもの……WAIT
- ② マシンサイクル単位で受付け、実行されるもの……リフレッシュ要求  
DMA要求  
BUSREQ
- ③ 命令単位で受付け、実行されるもの……各種割込み

優先順位は原則として

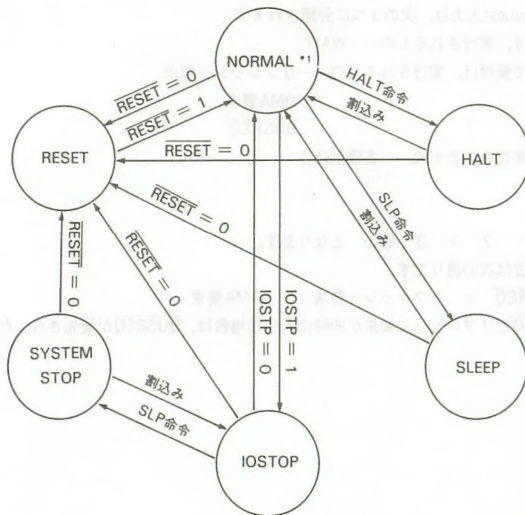
優先度 高い ① > ② > ③ 低い となります。

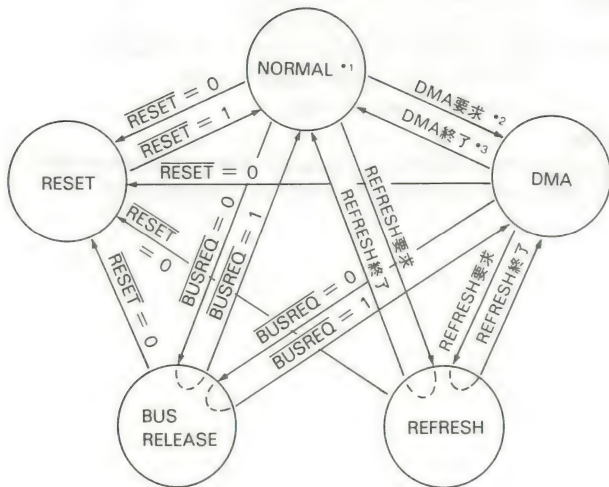
グループ②内の優先順位は次の通りです。

優先度 高い BUSREQ > リフレッシュ要求 > DMA要求

ただし、BUSREQとリフレッシュ要求が同時に入った場合は、BUSREQが優先され、かつリフレッシュ要求はクリアされます。

# 状態遷移図





注) \* 1 NORMALとは、通常のCPU命令実行状態を表わす。

\* 2 DMA要求:  $\overline{\text{DREQ}}_0, \overline{\text{DREQ}}_1 = 0$  (メモリ→(メモリマップ) I/O転送時)

または $\text{DE}0 = 1$  (メモリ→メモリ転送時)

\* 3 DMA終了:  $\overline{\text{DREQ}}_0, \overline{\text{DREQ}}_1 = 1$  (メモリ→(メモリマップ) I/O転送時)

または $\text{BCR}0, \text{BCR}1 = 0000\text{H}$  (全転送モード時)

または $\overline{\text{NMI}} = 0$  (全転送モード時)

#### 上図以外の状態遷移

1. HALT, IOSTOPからDMA, REFRESH, BUS RELEASEへの移行, およびその逆の移行可

2. SLEEP, SYSTEM STOPからBUS RELEASEへの移行, およびその逆の移行可

## ステータス信号一覧表

各モードでの状態信号の出力を、下記にまとめて示します。

モード		$\overline{\text{LIR}}$	$\overline{\text{ME}}$	$\overline{\text{IOE}}$	$\overline{\text{RD}}$	$\overline{\text{WR}}$	$\overline{\text{REF}}$	$\overline{\text{HALT}}$	$\overline{\text{BUSACK}}$	ST	Address BUS	Data BUS
CPU 動作	第1オペコードフェッチ	0	0	1	0	1	1	1	1	0	A	IN
	第1オペコード以外の オペコードフェッチ	0	0	1	0	1	1	1	1	1	A	IN
	メモリリード	1	0	1	0	1	1	1	1	1	A	IN
	メモリライト	1	0	1	1	0	1	1	1	1	A	OUT
	I/O リード	1	1	0	0	1	1	1	1	1	A	IN
	I/O ライト	1	1	0	1	0	1	1	1	1	A	OUT
	内部動作	1	1	1	1	1	1	1	1	1	A	IN
リフレッシュ		1	0	1	1	1	0	1	1	*	A	IN
割込み	$\overline{\text{NMI}}$	0	0	1	0	1	1	1	1	0	A	IN
アクノレッジ	$\overline{\text{INT}}_0$	0	1	0	1	1	1	1	1	0	A	IN
(第1マシン サイクル)	$\overline{\text{INT}}_1$ , $\overline{\text{INT}}_2$ , & 内部割込み	1	1	1	1	1	1	1	0	A	IN	

1 : Highレベル, 0 : Lowレベル, \* : 不定, A : 任意値, Z : ハイインピーダンス, IN : 入力, OUT : 出力

モード		$\overline{\text{LIR}}$	$\overline{\text{ME}}$	$\overline{\text{IOE}}$	$\overline{\text{RD}}$	$\overline{\text{WR}}$	$\overline{\text{REF}}$	$\overline{\text{HALT}}$	$\overline{\text{BUSACK}}$	ST	Address BUS	Data BUS
BUS RELEASE		1	Z	Z	Z	Z	1	1	0	*	Z	IN
HALT		0	0	1	0	1	1	0	1	0	A	IN
SLEEP		1	1	1	1	1	1	0	1	1	1	IN
内部 DMA	メモリリード	1	0	1	0	1	1	1	1	0	A	IN
	メモリライト	1	0	1	1	0	1	1	1	0	A	OUT
	I/O リード	1	1	0	0	1	1	1	1	0	A	IN
	I/O ライト	1	1	0	1	0	1	1	1	0	A	OUT
リセット		1	1	1	1	1	1	1	1	1	Z	IN

1 : Highレベル, 0 : Lowレベル, \* : 不定, A : 任意値, Z : ハイインピーダンス, IN : 入力, OUT : 出力

# リセット時および低消費電力モード時の各端子の状態

記号	端子のモード選択	端 子 の 状 態			
		リセット時	SLEEPモード時	IOSTOPモード時	SYSTEM STOPモード時
$\overline{\text{WAIT}}$	_____	IN (N)	IN (N)	IN (A)	IN (N)
$\overline{\text{BUSACK}}$	_____	1	OUT	OUT	OUT
$\overline{\text{BUSREQ}}$	_____	IN (N)	IN (A)	IN (A)	IN (A)
$\overline{\text{RESET}}$	_____	0	IN (A)	IN (A)	IN (A)
$\overline{\text{NMI}}$	_____	IN (N)	IN (A)	IN (A)	IN (A)
$\overline{\text{INT}}_0$	_____	IN (N)	IN (A)	IN (A)	IN (A)
$\overline{\text{INT}}_1$	_____	IN (N)	IN (A)	IN (A)	IN (A)
$\overline{\text{INT}}_2$	_____	IN (N)	IN (A)	IN (A)	IN (A)
ST	_____	1	1	OUT	1
$A_0 \sim A_{17}$	_____	Z	1	A	1
$A_{18}/\text{TOUT}$	$A_{18}$ 選択	Z	1	A	1
	TOUT選択	Z	OUT	H	H
$D_0 \sim D_7$	_____	Z	Z	A	Z

1 : Highレベル, 0 : Lowレベル, A : 任意値, Z : ハイインピーダンス, IN (A) : 入力(活性), IN (N) : 入力(非活性), OUT : 出力, H : 指示されたモードに移る前の状態を保持

記 号	端子のモード選択	端 子 の 状 態			
		リセット時	SLEEPモード時	IOSTOPモード時	SYSTEM STOPモード時
$\overline{\text{RTS}}_0$	—————	1	H	OUT	H
$\overline{\text{CTS}}_0$	—————	IN(N)	IN(A)	IN(N)	IN(N)
$\overline{\text{DCD}}_0$	—————	IN(N)	IN(A)	IN(N)	IN(N)
$\text{TXA}_0$	—————	1	OUT	H	H
$\text{RXA}_0$	—————	IN(N)	IN(A)	IN(N)	IN(N)
$\text{CKA}_0/\overline{\text{DREQ}}_0$	内部クロック選択	Z	OUT	Z	Z
	外部クロック選択	Z	IN(A)	IN(N)	IN(N)
	$\overline{\text{DREQ}}_0$ 選択	Z	IN(N)	IN(A)	IN(N)
$\text{TXA}_1$	—————	1	OUT	H	H
$\text{RXA}_1$	—————	IN(N)	IN(A)	IN(N)	IN(N)
$\text{CKA}_1/\overline{\text{TEND}}_0$	内部クロック選択	Z	OUT	Z	Z
	外部クロック選択	Z	IN(A)	IN(N)	IN(N)
	$\overline{\text{TEND}}_0$ 選択	Z	1	OUT	1
$\text{TXS}$	—————	1	OUT	H	H
$\text{RXS}/\overline{\text{CTS}}_1$	$\text{RXS}$ 選択	IN(N)	IN(A)	IN(N)	IN(N)
	$\overline{\text{CTS}}_1$ 選択	IN(N)	IN(A)	IN(N)	IN(N)



記 号	端子のモード選択	端 子 の 状 態			
		リセット時	SLEEPモード時	IOSTOPモード時	SYSTEM STOPモード時
CKS	内部クロック選択	Z	OUT	1	1
	外部クロック選択	Z	IN(A)	Z	Z
$\overline{\text{DREQ}}_1$	—————	IN(N)	IN(N)	IN(A)	IN(N)
$\overline{\text{TEND}}_1$	—————	1	1	OUT	1
$\overline{\text{HALT}}$	—————	1	OUT	OUT	0
$\overline{\text{REF}}$	—————	1	1	OUT	1
$\overline{\text{IOE}}$	—————	1	1	OUT	1
$\overline{\text{ME}}$	—————	1	1	OUT	1
E	—————	0	Eクロック出力	同 左	同 左
$\overline{\text{LIR}}$	—————	1	1	OUT	1
$\overline{\text{WR}}$	—————	1	1	OUT	1
$\overline{\text{RD}}$	—————	1	1	OUT	1
$\phi$	—————	$\phi$ クロック出力	同 左	同 左	同 左

1 : Highレベル, 0 : Lowレベル, A : 任意値, Z : ハイインピーダンス, IN(A) : 入力(活性), IN(N) : 入力(非活性), OUT : 出力, H : 指示されたモードに移る前の状態を保持

## 命令セット一覧

命令セットの一覧表の中で使用される記号を以下に説明します。

### 1. レジスタ指定

$g, g', ww, xx, yy, zz$  はレジスタを指定する記号です。 $g, g'$  は8ビットのレジスタ、 $ww, xx, yy, zz$  は16ビットのレジスタペアを指定します。各々対応するレジスタは下記の通りです。

$g, g'$	Reg.
000	B
001	C
010	D
011	E
100	H
101	L
111	A

$ww$	Reg.
00	BC
01	DE
10	HL
11	SP

$xx$	Reg.
00	BC
01	DE
10	IX
11	SP

$yy$	Reg.
00	BC
01	DE
10	IY
11	SP

$zz$	Reg.
00	BC
01	DE
10	HL
11	A,F

### 2. ビット指定

$b$  は、ビット操作命令におけるビットオペラントが何ビット目かを指定する記号です。各々対応するビットは下記の通りです。

$b$	Bit
000	0
001	1
010	2
011	3
100	4
101	5
110	6
111	7

### 3. コンディション指定

fは、演算の結果を判定して命令を実行する場合の条件を指定する記号です。各々対応するコンディションは下記の通りです。

f	Condition
000	NZ non zero
001	Z zero
010	NC non carry
011	C carry
100	PO parity odd
101	PE parity even
110	P sign plus
111	M sign minus

### 4. リスタートアドレス

vは、リスタート命令のリスタートアドレスを指定する記号です。各々対応するアドレスは下記の通りです。

v	Address
000	00H
001	08H
010	10H
011	18H
100	20H
101	28H
110	30H
111	38H

### 5. フラグ

フラグの変化を示す記号について説明します。

- ：その命令によってフラグは変化しない。
- ×：その命令によってフラグの変化は不定。
- ↑：その命令によって演算結果に従いフラグは変化。
- S：その命令によって“1”にセット。
- R：その命令によって“0”にリセット。
- P：その命令によってパリティフラグとして変化。
- V：その命令によってオーバーフローフラグとして変化。

### 6. その他

( )<sub>M</sub>：( )内の内容をアドレスとするメモリを表わします。

( )<sub>I</sub>：( )内の内容をアドレスとするI/Oを表わします。

m：8ビット値

mn：16ビット値

S：ソースアドレッシングモード

D：デスティネーションアドレッシングモード

b·( )<sub>M</sub> or b·gr：( )内の内容をアドレスとするメモリ、又はgrで指定されるレジスタの内容の、bで指定されるビットを表わします。

r or R：rの添字がついていると、8ビットレジスタを表わします。Rの添字がついていると、16ビットレジスタを表わします。

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag						
			S	D					7	6	4	2	1	0	
									S	Z	H	P/V	N	C	
ADD	ADD A, g	10 000 g	REG	IMP	2	1	4	$A_r + g_r \rightarrow A_r$	↑	↑	↑	V	R	↑	
	ADD A, (HL)	10 000 110	REGI	IMP	2	1	6	$A_r + (HL)_M \rightarrow A_r$	↑	↑	↑	V	R	↑	
	ADD A, m	11 000 110 < m >	IMMED	IMP	2	2	6	$A_r + m \rightarrow A_r$	↑	↑	↑	V	R	↑	
	ADD A, (IX+d)	11 011 101 10 000 110 < d >	IND	IMP	6	3	14	$A_r + (IX+d)_M \rightarrow A_r$	↑	↑	↑	V	R	↑	
	ADD A, (IY+d)	11 111 101 10 000 110 < d >	IND	IMP	6	3	14	$A_r + (IY+d)_M \rightarrow A_r$	↑	↑	↑	V	R	↑	
ADC	ADC A, g	10 001 g	REG	IMP	2	1	4	$A_r + g_r + c \rightarrow A_r$	↑	↑	↑	V	R	↑	
	ADC A, (HL)	10 001 110	REGI	IMP	2	1	6	$A_r + (HL)_M + c \rightarrow A_r$	↑	↑	↑	V	R	↑	
	ADC A, m	11 001 110 < m >	IMMED	IMP	2	2	6	$A_r + m + c \rightarrow A_r$	↑	↑	↑	V	R	↑	
	ADC A, (IX+d)	11 011 101 10 001 110 < d >	IND	IMP	6	3	14	$A_r + (IX+d)_M + c \rightarrow A_r$	↑	↑	↑	V	R	↑	
	ADC A, (IY+d)	11 111 101 10 001 110 < d >	IND	IMP	6	3	14	$A_r + (IY+d)_M + c \rightarrow A_r$	↑	↑	↑	V	R	↑	

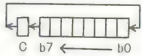
Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
									7	6	4	2	1	0
			S	D					S	Z	H	P/V	N	C
AND	AND g	10 100 g	REG	IMP	2	1	4	$A_r \cdot g_r \rightarrow A_r$	$\uparrow$	$\uparrow$	S	P	R	R
	AND (HL)	10 100 110	REGI	IMP	2	1	6	$A_r \cdot (HL)_M \rightarrow A_r$	$\uparrow$	$\uparrow$	S	P	R	R
	AND m	11 100 110 < m >	IMMED	IMP	2	2	6	$A_r \cdot m \rightarrow A_r$	$\uparrow$	$\uparrow$	S	P	R	R
	AND (IX+d)	11 011 101 10 100 110 < d >	IND	IMP	6	3	14	$A_r \cdot (IX+d)_M \rightarrow A_r$	$\uparrow$	$\uparrow$	S	P	R	R
	AND (IY+d)	11 111 101 10 100 110 < d >	IND	IMP	6	3	14	$A_r \cdot (IY+d)_M \rightarrow A_r$	$\uparrow$	$\uparrow$	S	P	R	R
Compare	CP g	10 111 g	REG	IMP	2	1	4	$A_r - g_r$	$\uparrow$	$\uparrow$	$\uparrow$	V	S	$\uparrow$
	CP (HL)	10 111 110	REGI	IMP	2	1	6	$A_r - (HL)_M$	$\uparrow$	$\uparrow$	$\uparrow$	V	S	$\uparrow$
	CP m	11 111 110 < m >	IMMED	IMP	2	2	6	$A_r - m$	$\uparrow$	$\uparrow$	$\uparrow$	V	S	$\uparrow$
	CP (IX+d)	11 011 101 10 111 110 < d >	IND	IMP	6	3	14	$A_r - (IX+d)_M$	$\uparrow$	$\uparrow$	$\uparrow$	V	S	$\uparrow$
	CP (IY+d)	11 111 101 10 111 110 < d >	IND	IMP	6	3	14	$A_r - (IY+d)_M$	$\uparrow$	$\uparrow$	$\uparrow$	V	S	$\uparrow$

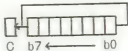
Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag							
			S	D					7	6	4	2	1	0		
									S	Z	H	P/V	N	C		
COMPLEMENT	CPL	00 101 111	IMP	IMP	1	1	3	$\overline{A_r} \rightarrow A_r$	●	●	S	●	S	●		
DEC	DEC g	00 g 101	REG	REG	2	1	4	$g_r - 1 \rightarrow g_r$	↑	↑	↑	V	S	●		
	DEC (HL)	00 110 101	REG1	REG1	4	1	10	$(HL)_M - 1 \rightarrow (HL)_M$	↑	↑	↑	V	S	●		
	DEC (IX + d)	11 011 101	IND	IND	8	3	18	$(IX + d)_M - 1 \rightarrow (IX + d)_M$	↑	↑	↑	V	S	●		
		00 110 101 < d >														
	DEC (IY + d)	11 111 101	IND	IND	8	3	18	$(IY + d)_M - 1 \rightarrow (IY + d)_M$	↑	↑	↑	V	S	●		
		00 110 101 < d >														
INC	INC g	00 g 100	REG	REG	2	1	4	$g_r + 1 \rightarrow g_r$	↑	↑	↑	V	R	●		
	INC (HL)	00 110 100	REG1	REG1	4	1	10	$(HL)_M + 1 \rightarrow (HL)_M$	↑	↑	↑	V	R	●		
	INC (IX + d)	11 011 101	IND	IND	8	3	18	$(IX + d)_M + 1 \rightarrow (IX + d)_M$	↑	↑	↑	V	R	●		
		00 110 100 < d >														
	INC (IY + d)	11 111 101	IND	IND	8	3	18	$(IY + d)_M + 1 \rightarrow (IY + d)_M$	↑	↑	↑	V	R	●		
		00 110 100 < d >														
MULT	MLT ww	11 101 101 01 ww1 100	REG	REG	13	2	17	$wwH_r \times wwL_r \rightarrow wwR$	●	●	●	●	●	●		


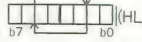



Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
			S	D					7	6	4	2	1	0
NEGATE	NEG	11 101 101 01 000 100	IMP	IMP	2	2	6	$0 - A_r \rightarrow A_r$	↑	↑	↑	V	S	↑
OR	OR g	10 110 g	REG	IMP	2	1	4	$A_r + g_r \rightarrow A_r$	↑	↑	R	P	R	R
	OR (HL)	10 110 110	REGI	IMP	2	1	6	$A_r + (HL)_M \rightarrow A_r$	↑	↑	R	P	R	R
	OR m	11 110 110 < m >	IMMED	IMP	2	2	6	$A_r + m \rightarrow A_r$	↑	↑	R	P	R	R
	OR (IX + d)	11 011 101 10 110 110 < d >	IND	IMP	6	3	14	$A_r + (IX + d)_M \rightarrow A_r$	↑	↑	R	P	R	R
	OR (IY + d)	11 111 101 10 110 110 < d >	IND	IMP	6	3	14	$A_r + (IY + d)_M \rightarrow A_r$	↑	↑	R	P	R	R
SUB	SUB g	10 010 g	REG	IMP	2	1	4	$A_r - g_r \rightarrow A_r$	↑	↑	↑	V	S	↑
	SUB (HL)	10 010 110	REGI	IMP	2	1	6	$A_r - (HL)_M \rightarrow A_r$	↑	↑	↑	V	S	↑
	SUB m	11 010 110 < m >	IMMED	IMP	2	2	6	$A_r - m \rightarrow A_r$	↑	↑	↑	V	S	↑
	SUB (IX + d)	11 011 101 10 010 110 < d >	IND	IMP	6	3	14	$A_r - (IX + d)_M \rightarrow A_r$	↑	↑	↑	V	S	↑

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
									7	6	4	2	1	0
			S	D					S	Z	H	P/V	N	C
SUB	SUB (IY + d)	11 111 101 10 010 110 < d >	IND	IMP	6	3	14	$A_r - (IY + d)_M \rightarrow A_r$	↑	↑	↑	V	S	↑
SUBC	SBC A, g	10 011 g	REG	IMP	2	1	4	$A_r - g_r - c \rightarrow A_r$	↑	↑	↑	V	S	↑
	SBC A, (HL)	10 011 110	REGI	IMP	2	1	6	$A_r - (HL)_M - c \rightarrow A_r$	↑	↑	↑	V	S	↑
	SBC A, m	11 011 110 < m >	IMMED	IMP	2	2	6	$A_r - m - c \rightarrow A_r$	↑	↑	↑	V	S	↑
	SBC A, (IX + d)	11 011 101 10 011 110 < d >	IND	IMP	6	3	14	$A_r - (IX + d)_M - c \rightarrow A_r$	↑	↑	↑	V	S	↑
	SBC A, (IY + d)	11 111 101 10 011 110 < d >	IND	IMP	6	3	14	$A_r - (IY + d)_M - c \rightarrow A_r$	↑	↑	↑	V	S	↑
TEST	TST g	11 101 101 00 g 100	REG		3	2	7	$A_r \cdot g_r$	↑	↑	S	P	R	R
	TST (HL)	11 101 101 00 110 100	REGI		4	2	10	$A_r \cdot (HL)_M$	↑	↑	S	P	R	R

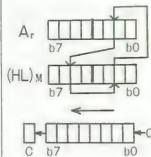



Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
									7	6	4	2	1	0
			S	D					S	Z	H	P/V	N	C
TEST	TST m	11 101 101 01 100 100 < m >	IMMED		3	3	9	$A_r \cdot m$	↑	↑	S	P	R	R
XOR	XOR g	10 101 g	REG	IMP	2	1	4	$A_r \oplus g_r \rightarrow A_r$	↑	↑	R	P	R	R
	XOR (HL)	10 101 110	REGI	IMP	2	1	6	$A_r \oplus (HL)_M \rightarrow A_r$	↑	↑	R	P	R	R
	XOR m	11 101 110 < m >	IMMED	IMP	2	2	6	$A_r \oplus m \rightarrow A_r$	↑	↑	R	P	R	R
	XOR (IX + d)	11 011 101 10 101 110 < d >	IND	IMP	6	3	14	$A_r \oplus (IX + d)_M \rightarrow A_r$	↑	↑	R	P	R	R
	XOR (IY + d)	11 111 101 10 101 110 < d >	IND	IMP	6	3	14	$A_r \oplus (IY + d)_M \rightarrow A_r$	↑	↑	R	P	R	R
Rotate and Shift Data	RLA	00 010 111	IMP	IMP	1	1	3		●	●	R	●	R	↑
	RL g	11 001 011	REG	REG	3	2	7		↑	↑	R	P	R	↑
	RL (HL)	00 010 g 11 001 011 00 010 110	REGI	REGI	5	2	13		↑	↑	R	P	R	↑

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
									7	6	4	2	1	0
			S	D					S	Z	H	P/V	N	C
Rotate and Shift Data	RL (IX + d)	11 011 101	IND	IND	7	4	19		↑	↑	R	P	R	↑
		11 001 011 < d >												
		00 010 110												
	RL (IY + d)	11 111 101	IND	IND	7	4	19		↑	↑	R	P	R	↑
		11 001 011 < d >												
		00 010 110												
	RLCA	00 000 111	IMP	IMP	1	1	3		●	●	R	●	R	↑
	RLC g	11 001 011	REG	REG	3	2	7		↑	↑	R	P	R	↑
		00 000 g												
	RLC (HL)	11 001 011	REGI	REGI	5	2	13		↑	↑	R	P	R	↑
		00 000 110												
	RLC (IX + d)	11 011 101	IND	IND	7	4	19		↑	↑	R	P	R	↑
		11 001 011 < d >												
		00 000 110												
	RLC (IY + d)	11 111 101	IND	IND	7	4	19		↑	↑	R	P	R	↑
		11 001 011 < d >												
		00 000 110												

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
									7	6	4	2	1	0
			S	D					S	Z	H	P/V	N	C
Rotate and Shift Data	RLD	11 101 101 01 101 111	IMP	IMP	8	2	16		↑	↑	R	P	R	●
	RRA	00 011 111	IMP	IMP	1	1	3		●	●	R	●	R	↑
	RR g	11 001 011 00 011 g	REG	REG	3	2	7		↑	↑	R	P	R	↑
	RR (HL)	11 001 011 00 011 110	REGI	REGI	5	2	13		↑	↑	R	P	R	↑
	RR (IX+d)	11 011 101 11 001 011 < d >	IND	IND	7	4	19		↑	↑	R	P	R	↑
	RR (IY+d)	00 011 110 11 111 101 11 001 011 < d >	IND	IND	7	4	19		↑	↑	R	P	R	↑
	RRCA	00 001 111	IMP	IMP	1	1	3		●	●	R	●	R	↑
	RRC g	11 001 011 00 001 g	REG	REG	3	2	7		↑	↑	R	P	R	↑
	RRC (HL)	11 001 011 00 001 110	REGI	REGI	5	2	13		↑	↑	R	P	R	↑

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
									7	6	4	2	1	0
			S	D					S	Z	P/V	H	N	C
Rotate and Shift Data	RRC (IX+d)	11 011 101	IND	IND	7	4	19							↑ ↑ R P R ↑
		11 001 011												
		< d >												
	RRC (IY+d)	00 001 110	IND	IND	7	4	19							↑ ↑ R P R ↑
		11 111 101												
		11 001 011												
	RRD	< d >	IMP	IMP	8	2	16							↑ ↑ R P R ●
		00 001 110												
		11 101 101												
	SLA g	01 100 111	REG	REG	3	2	7							↑ ↑ R P R ↑
		00 100 g												
	SLA (HL)	11 001 011	REGI	REGI	5	2	13							↑ ↑ R P R ↑
		00 100 110												
	SLA (IX+d)	11 011 101	IND	IND	7	4	19							↑ ↑ R P R ↑
		11 001 011												
		< d >												
	SLA (IY+d)	00 100 110	IND	IND	7	4	19							↑ ↑ R P R ↑
		11 111 101												
		11 001 011												
	SRA g	< d >	REG	REG	3	2	7							↑ ↑ R P R ↑
		00 100 110												
		11 001 011												
		00 101 g												



Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
			S	D					7	6	4	2	1	0
									S	Z	H	P/V	N	C
Rotate and Shift Data	SRA (HL)	11 001 011	REGI	REGI	5	2	13		↑	↑	R	P	R	↓
		00 101 110												
	SRA (IX + d)	11 011 101	IND	IND	7	4	19		↑	↑	R	P	R	↑
		11 001 011 < d >												
	SRA (IY + d)	00 101 110	IND	IND	7	4	19		↑	↑	R	P	R	↑
		11 111 101 11 001 011 < d >												
	SRL g	00 101 110	REG	REG	3	2	7		↑	↑	R	P	R	↑
		11 001 011 00 111 g												
	SRL (HL)	11 001 011	REGI	REGI	5	2	13		↑	↑	R	P	R	↑
		00 111 110												
	SRL (IX + d)	11 011 101	IND	IND	7	4	19		↑	↑	R	P	R	↑
		11 001 011 < d >												
	SRL (IY + d)	00 111 110	IND	IND	7	4	19		↑	↑	R	P	R	↑
		11 111 101 11 001 011 < d >												
		00 111 110												

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
									7	6	4	2	1	0
			S	D					S	Z	H	P/V	N	C
Bit Set	SET b, g	11 001 011 11 b g	REG	REG	3	2	7	$1 \rightarrow b \cdot g_r$	●	●	●	●	●	●
	SET b, (HL)	11 001 011 11 b 110	REGI	REGI	5	2	13	$1 \rightarrow b \cdot (HL)_M$	●	●	●	●	●	●
	SET b, (IX + d)	11 011 101 11 001 011 < d >	IND	IND	7	4	19	$1 \rightarrow b \cdot (IX + d)_M$	●	●	●	●	●	●
	SET b, (IY + d)	11 001 011 11 b 110 11 111 101 11 001 011 < d > 11 b 110	IND	IND	7	4	19	$1 \rightarrow b \cdot (IY + d)_M$	●	●	●	●	●	●
Bit Reset	RES b, g	11 001 011 10 b g	REG	REG	3	2	7	$0 \rightarrow b \cdot g_r$	●	●	●	●	●	●
	RES b, (HL)	11 001 011 10 b 110	REGI	REGI	5	2	13	$0 \rightarrow b \cdot (HL)_M$	●	●	●	●	●	●
	RES b, (IX + d)	11 011 101 11 001 011 < d >	IND	IND	7	4	19	$0 \rightarrow b \cdot (IX + d)_M$	●	●	●	●	●	●
		10 b 110												

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
									7	6	4	2	1	0
			S	D					S	Z	H	P/V	N	C
Bit Reset	RES b,(IY + d)	11 111 101 11 001 011 < d > 10 b   110	IND	IND	7	4	19	$0 \rightarrow b \cdot (IY + d)_M$	●	●	●	●	●	●
Bit Test	BIT b, g	11 001 011 01 b   g	REG		2	2	6	$\overline{b \cdot g_r} \rightarrow z$	X	↑	S	X	R	●
	BIT b,(HL)	11 001 011 01 b   110	REGI		3	2	9	$\overline{b \cdot (HL)_M} \rightarrow z$	X	↑	S	X	R	●
	BIT b,(IX + d)	11 011 101 11 001 011 < d > 01 b   110	IND		5	4	15	$\overline{b \cdot (IX + d)_M} \rightarrow z$	X	↑	S	X	R	●
	BIT b,(IY + d)	11 111 101 11 001 011 < d > 01 b   110	IND		5	4	15	$\overline{b \cdot (IY + d)_M} \rightarrow z$	X	↑	S	X	R	●

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
									7	6	4	2	1	0
			S	D					S	Z	H	P/V	N	C
ADD	ADD HL, ww	00 ww1 001	REG	IMP	5	1	7	$HL_R + ww_R \rightarrow HL_R$	●	●	X	●	R	↑
	ADD IX, xx	11 011 101	REG	IMP	6	2	10	$IX_R + xx_R \rightarrow IX_R$	●	●	X	●	R	↑
		00 xx1 001												
	ADD IY, yy	11 111 101	REG	IMP	6	2	10	$IY_R + yy_R \rightarrow IY_R$	●	●	X	●	R	↑
		00 yy1 001												
ADC	ADC HL, ww	11 101 101 01 ww1 010	REG	IMP	6	2	10	$HL_R + ww_R + c \rightarrow HL_R$	↑	↑	X	V	R	↑
DEC	DEC ww	00 ww1 011	REG	REG	2	1	4	$ww_R - 1 \rightarrow ww_R$	●	●	●	●	●	●
	DEC IX	11 011 101	IMP	IMP	3	2	7	$IX_R - 1 \rightarrow IX_R$	●	●	●	●	●	●
		00 101 011												
	DEC IY	11 111 101	IMP	IMP	3	2	7	$IY_R - 1 \rightarrow IY_R$	●	●	●	●	●	●
		00 101 011												
INC	INC ww	00 ww0 011	REG	REG	2	1	4	$ww_R + 1 \rightarrow ww_R$	●	●	●	●	●	●
	INC IX	11 011 101	IMP	IMP	3	2	7	$IX_R + 1 \rightarrow IX_R$	●	●	●	●	●	●
		00 100 011												
	INC IY	11 111 101	IMP	IMP	3	2	7	$IY_R + 1 \rightarrow IY_R$	●	●	●	●	●	●
		00 100 011												



Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag
									7 6 4 2 1 0
			S	D					S Z H P/V N C
SBC	SBC HL, ww	11 101 101 01 ww0 010	REG	IMP	6	2	10	$HL_R - ww_R - C \rightarrow HL_R$	$\uparrow \downarrow X V S \downarrow$
Load 8 bit Data	LD A, I	11 101 101 01 010 111	IMP	IMP	2	2	6	$I_r \rightarrow A_r$	$\uparrow \downarrow R IEF_2 R \bullet$
	LD A, R	11 101 101 01 011 111	IMP	IMP	2	2	6	$R_r \rightarrow A_r$	$\uparrow \downarrow R IEF_2 R \bullet$
	LD A, (BC)	00 001 010	REGI	IMP	2	1	6	$(BC)_M \rightarrow A_r$	$\bullet \bullet \bullet \bullet \bullet \bullet$
	LD A, (DE)	00 011 010	REGI	IMP	2	1	6	$(DE)_M \rightarrow A_r$	$\bullet \bullet \bullet \bullet \bullet \bullet$
	LD A, (mn)	00 111 010 < n > < m >	EXT	IMP	4	3	12	$(mn)_M \rightarrow A_r$	$\bullet \bullet \bullet \bullet \bullet \bullet$
	LD I, A	11 101 101 01 000 111	IMP	IMP	2	2	6	$A_r \rightarrow I_r$	$\bullet \bullet \bullet \bullet \bullet \bullet$
	LD R, A	11 101 101 01 001 111	IMP	IMP	2	2	6	$A_r \rightarrow R_r$	$\bullet \bullet \bullet \bullet \bullet \bullet$
	LD (BC), A	00 000 010	IMP	REGI	3	1	7	$A_r \rightarrow (BC)_M$	$\bullet \bullet \bullet \bullet \bullet \bullet$
	LD (DE), A	00 010 010	IMP	REGI	3	1	7	$A_r \rightarrow (DE)_M$	$\bullet \bullet \bullet \bullet \bullet \bullet$
	LD (mn), A	00 110 010 < n > < m >	IMP	EXT	5	3	13	$A_r \rightarrow (mn)_M$	$\bullet \bullet \bullet \bullet \bullet \bullet$

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
									7	6	4	2	1	0
			S	D					S	Z	H	P/V	N	C
Load 8 Bit Data	LD g, g'	01 g g'	REG	REG	2	1	4	$g_r' \rightarrow g_r$	●	●	●	●	●	●
	LD g, (HL)	01 g 110	REGI	REG	2	1	6	$(HL)_M \rightarrow g_r$	●	●	●	●	●	●
	LD g, m	00 g 110	IMMED	REG	2	2	6	$m \rightarrow g_r$	●	●	●	●	●	●
		< m >												
	LD g, (IX + d)	11 011 101	IND	REG	6	3	14	$(IX + d)_M \rightarrow g_r$	●	●	●	●	●	●
		01 g 110												
		< d >												
	LD g, (IY + d)	11 111 101	IND	REG	6	3	14	$(IY + d)_M \rightarrow g_r$	●	●	●	●	●	●
		01 g 110												
		< d >												
	LD (HL), m	00 110 110	IMMED	REGI	3	2	9	$m \rightarrow (HL)_M$	●	●	●	●	●	●
		< m >												
	LD (IX + d), m	11 011 101	IMMED	IND	5	4	15	$m \rightarrow (IX + d)_M$	●	●	●	●	●	●
		00 110 110												
		< d >												
		< m >												
	LD (IY + d), m	11 111 101	IMMED	IND	5	4	15	$m \rightarrow (IY + d)_M$	●	●	●	●	●	●
		00 110 110												
		< d >												
		< m >												
	LD (HL), g	01 110 g	REG	REGI	3	1	7	$g_r \rightarrow (HL)_M$	●	●	●	●	●	●

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles			Operation	Flag					
			S	D					7	6	4	2	1	0
Load 8Bit Data	LD (IX+d), g	11 011 101 01 110 g < d >	REG	IND	7	3	15	$g_r \rightarrow (IX+d)_M$	•	•	•	•	•	•
	LD (IY+d), g	11 111 101 01 110 g < d >	REG	IND	7	3	15	$g_r \rightarrow (IY+d)_M$	•	•	•	•	•	•
Load 16Bit Data	LD ww, mn	00 ww0 001 < n > < m >	IMMED	REG	3	3	9	$mn \rightarrow ww_R$	•	•	•	•	•	•
	LD IX, mn	11 011 101 00 100 001 < n > < m >	IMMED	IMP	4	4	12	$mn \rightarrow IX_R$	•	•	•	•	•	•
	LD IY, mn	11 111 101 00 100 001 < n > < m >	IMMED	IMP	4	4	12	$mn \rightarrow IY_R$	•	•	•	•	•	•
	LD SP, HL	11 111 001	IMP	IMP	2	1	4	$HL_R \rightarrow SP_R$	•	•	•	•	•	•
	LD SP, IX	11 011 101	IMP	IMP	3	2	7	$IX_R \rightarrow SP_R$	•	•	•	•	•	•
		11 111 001												

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
									7	6	4	2	1	0
			S	D					S	Z	H	P/V	N	C
Load 16Bit Data	LD SP, IY	11 111 101	IMP	IMP	3	2	7	$IY_R \rightarrow SP_R$	●	●	●	●	●	●
		11 111 001												
	LD ww, (mn)	11 101 101	EXT	REG	6	4	18	$(mn+1)_M \rightarrow wwH_r$ $(mn)_M \rightarrow wwL_r$	●	●	●	●	●	●
		01 ww1 011												
		< n >												
	LD HL, (mn)	< m >												
		00 101 010	EXT	IMP	5	3	15	$(mn+1)_M \rightarrow H_r$ $(mn)_M \rightarrow L_r$	●	●	●	●	●	●
		< n >												
		< m >												
	LD IX, (mn)	11 011 101	EXT	IMP	6	4	18	$(mn+1)_M \rightarrow IXH_r$ $(mn)_M \rightarrow IXL_r$	●	●	●	●	●	●
		00 101 010												
		< n >												
	LD IY, (mn)	< m >												
		11 111 101	EXT	IMP	6	4	18	$(mn+1)_M \rightarrow IYH_r$ $(mn)_M \rightarrow IYL_r$	●	●	●	●	●	●
		00 101 010												
		< n >												
	LD (mn), ww	< m >	REG	EXT	7	4	19	$wwH_r \rightarrow (mn+1)_M$ $wwL_r \rightarrow (mn)_M$	●	●	●	●	●	●
		11 101 101												
		01 ww0 011												
		< n >												
		< m >												

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
			S	D					7	6	4	2	1	0
									S	Z	H	P/V	N	C
Load 16bit Data	LD (mn), HL	00 100 010 < n > < m >	IMP	EXT	6	3	16	$H_r \rightarrow (mn+1)_M$ $L_r \rightarrow (mn)_M$	●	●	●	●	●	●
	LD (mn), IX	11 011 101 00 100 010 < n > < m >	IMP	EXT	7	4	19	$IXH_r \rightarrow (mn+1)_M$ $IXL_r \rightarrow (mn)_M$	●	●	●	●	●	●
	LD (mn), IY	11 111 101 00 100 010 < n > < m >	IMP	EXT	7	4	19	$IYH_r \rightarrow (mn+1)_M$ $IYL_r \rightarrow (mn)_M$	●	●	●	●	●	●
Block Transfer Search Data	CPD	11 101 101 10 101 001	REGI/ IMP		6	2	12	$A_r - (HL)_M$ $BC_R - 1 \rightarrow BC_R$ $HL_R - 1 \rightarrow HL_R$	② ↑	① ↑	↑	↑	S	●
	CPDR	11 101 101 10 111 001	REGI/ IMP		8 6	2 2	14 12	$BC_R \neq 0 \ A_r \neq (HL)_M$ $BC_R = 0 \text{ or } A_r = (HL)_M$ $\left[ \begin{array}{l} A_r - (HL)_M \\ BC_R - 1 \rightarrow BC_R \\ HL_R - 1 \rightarrow HL_R \end{array} \right]$ Repeat Q until $A_r = (HL)_M \text{ or } BC_R = 0$	② ↑	① ↑	↑	↑	S	●

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
									7	6	4	2	1	0
			S	D					S	Z	H	P/V	N	C
Block Transfer Search Data	CPI	11 101 101	REGI/ IMP		6	2	12	$A_r - (HL)_M$	②	①				
		10 100 001						$BC_R - 1 \rightarrow BC_R$ $HL_R + 1 \rightarrow HL_R$	↑	↑	↑	↑	S	●
	CPIR	11 101 101	REGI/ IMP		8	2	14	$BC_R \neq 0 \ A_r \neq (HL)_M$	②	①				
		10 110 001						$BC_R = 0$ or $A_r = (HL)_M$	↑	↑	↑	↑	S	●
	LDD	11 101 101 10 101 000	REGI	REGI	4	2	12	$Q \begin{cases} A_r - (HL)_M \\ BC_R - 1 \rightarrow BC_R \\ HL_R + 1 \rightarrow HL_R \end{cases}$ Repeat Q until $A_r = (HL)_M$ or $BC_R = 0$ $(HL)_M \rightarrow (DE)_M$ $BC_R - 1 \rightarrow BC_R$ $DE_R - 1 \rightarrow DE_R$ $HL_R - 1 \rightarrow HL_R$						
											①			

- ①  $P/V = 0 : BC_R - 1 = 0$   
 $P/V = 1 : BC_R - 1 \neq 0$
- ②  $Z = 1 : A_r = (HL)_M$   
 $Z = 0 : A_r \neq (HL)_M$

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag						
			S	D					7	6	4	2	1	0	
									S	Z	H	P/V	N	C	
Block Transfer Search Data	LDDR	11 101 101 10 111 000	REGI	REGI	6	2	14 (BC <sub>R</sub> ≠0)	Q [ (HL) <sub>M</sub> →(DE) <sub>M</sub> BC <sub>R</sub> -1→BC <sub>R</sub> DE <sub>R</sub> -1→DE <sub>R</sub> HL <sub>R</sub> -1→HL <sub>R</sub>	●	●	R	R	R	●	
					4	2	12 (BC <sub>R</sub> =0)								
	LDI	11 101 101 10 100 000	REGI	REGI	4	2	12	Repeat Q until BC <sub>R</sub> =0 (HL) <sub>M</sub> →(DE) <sub>M</sub> BC <sub>R</sub> -1→BC <sub>R</sub> DE <sub>R</sub> +1→DE <sub>R</sub> HL <sub>R</sub> +1→HL <sub>R</sub>	●	●	R	① ↑	R	●	
					LDIR	11 101 101 10 110 000	REGI	REGI							6
	4	2	12 (BC <sub>R</sub> =0)												
									Repeat Q until BC <sub>R</sub> =0						
PUSH	PUSH zz	11 zz0 101	REG	IMP	5	1	11	zzL <sub>r</sub> →(SP-2) <sub>M</sub> zzH <sub>r</sub> →(SP-1) <sub>M</sub> SP <sub>R</sub> -2→SP <sub>R</sub>	●	●	●	●	●	●	●

① P/V=0: BC<sub>R</sub>-1=0, P/V=1: BC<sub>R</sub>-1≠0

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes		Operation	Flag					
									7	6	4	2	1	0
			S	D					S	Z	H	P/V	N	C
PUSH	PUSH IX	11 011 101 11 100 101	IMP	IMP	6	2	14	IXL <sub>r</sub> → (SP - 2) <sub>M</sub> IXH <sub>r</sub> → (SP - 1) <sub>M</sub> SP <sub>R</sub> - 2 → SP <sub>R</sub>	●	●	●	●	●	●
	PUSH IY	11 111 101 11 100 101	IMP	IMP	6	2	14	IYL <sub>r</sub> → (SP - 2) <sub>M</sub> IYH <sub>r</sub> → (SP - 1) <sub>M</sub> SP <sub>R</sub> - 2 → SP <sub>R</sub>	●	●	●	●	●	●
POP	POP zz	11 zz0 001	IMP	REG	3	1	9	(SP + 1) <sub>M</sub> → zzH <sub>r</sub> (SP) <sub>M</sub> → zzL <sub>r</sub> SP <sub>R</sub> + 2 → SP <sub>R</sub>	●	●	●	●	●	●
	POP IX	11 011 101 11 100 001	IMP	IMP	4	2	12	(SP + 1) <sub>M</sub> → IXH <sub>r</sub> (SP) <sub>M</sub> → IXL <sub>r</sub> SP <sub>R</sub> + 2 → SP <sub>R</sub>	●	●	●	●	●	●
	POP IY	11 111 101 11 100 001	IMP	IMP	4	2	12	(SP + 1) <sub>M</sub> → IYH <sub>r</sub> (SP) <sub>M</sub> → IYL <sub>r</sub> SP <sub>R</sub> + 2 → SP <sub>R</sub>	●	●	●	●	●	●
Exchange	EX AF, AF	00 001 000	IMP	IMP	2	1	4	AF <sub>R</sub> ↔ AF' <sub>R</sub>	●	●	●	●	●	●
	EX DE, HL	11 101 011	IMP	IMP	1	1	3	DE <sub>R</sub> ↔ HL <sub>R</sub>	●	●	●	●	●	●
	EXX	11 011 001	IMP	IMP	1	1	3	BC <sub>R</sub> ↔ BC' <sub>R</sub>	●	●	●	●	●	●
								DE <sub>R</sub> ↔ DE' <sub>R</sub> HL <sub>R</sub> ↔ HL' <sub>R</sub>						



Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
			S	D					7	6	4	2	1	0
									S	Z	H	P/V	N	C
Exchange	EX (SP), HL	11 100 011	IMP	IMP	6	1	16	$H_r \leftrightarrow (SP+1)_M$ $L_r \leftrightarrow (SP)_M$	●	●	●	●	●	●
	EX (SP), IX	11 011 101 11 100 011	IMP	IMP	7	2	19	$IXH_r \leftrightarrow (SP+1)_M$ $IXL_r \leftrightarrow (SP)_M$	●	●	●	●	●	●
	EX (SP), IY	11 111 101 11 100 011	IMP	IMP	7	2	19	$IYH_r \leftrightarrow (SP+1)_M$ $IYL_r \leftrightarrow (SP)_M$	●	●	●	●	●	●
Call	CALL mn	11 001 101 < n > < m >		EXT	6	3	16	$PCH_r \rightarrow (SP-1)_M$ $PCL_r \rightarrow (SP-2)_M$ $mn \rightarrow PC_R$ $SP_R - 2 \rightarrow SP_R$	●	●	●	●	●	●
	CALL f, mn	11 f 100 < n > < m >		EXT	2	3	6(f : false)	continue : f is false	●	●	●	●	●	●
					6	3	16(f : true)	CALL mn : f is true						
Jump	DJNZ j	00 010 000 < j-2 >		REL	5 3	2 2	9 7	$(B_r \neq 0)$ $(B_r = 0)$ $B_r - 1 \rightarrow B_r$ contine : $B_r = 0$ $PC_R + j \rightarrow PC_R$ : $B_r \neq 0$	●	●	●	●	●	●

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
			S	D					7	6	4	2	1	0
									S	Z	H	P/V	N	C
Jump	JP f, mn	11 f 010		EXT	2	3	6	(f : false)	●	●	●	●	●	●
		< n >			3	3	9	(f : true)						
		< m >						mn→PC <sub>R</sub> : f is true						
	JP mn	11 000 011		EXT	3	3	9	continue : f is false						
		< n >						mn→PC <sub>R</sub>	●	●	●	●	●	●
		< m >												
	JP (HL)	11 101 001		REGI	1	1	3	HL <sub>R</sub> →PC <sub>R</sub>	●	●	●	●	●	●
	JP (IX)	11 011 101		REGI	2	2	6	IX <sub>R</sub> →PC <sub>R</sub>	●	●	●	●	●	●
	JP (IY)	11 101 001		REGI	2	2	6	IY <sub>R</sub> →PC <sub>R</sub>	●	●	●	●	●	●
		11 111 101												
	JR j	00 011 000		REL	4	2	8	PC <sub>R</sub> +j→PC <sub>R</sub>	●	●	●	●	●	●
	< j-2 >													
JR C, j	00 111 000		REL	2	2	6	continue : C=0	●	●	●	●	●	●	
< j-2 >	4			2	8	PC <sub>R</sub> +j→PC <sub>R</sub> : C=1								
JR NC, j	00 110 000		REL	2	2	6	continue : C=1	●	●	●	●	●	●	
< j-2 >	4			2	8	PC <sub>R</sub> +j→PC <sub>R</sub> : C=0								
JR Z, j	00 101 000		REL	2	2	6	continue : Z=0	●	●	●	●	●	●	
< j-2 >	4			2	8	PC <sub>R</sub> +j→PC <sub>R</sub> : Z=1								

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
			S	D					7	6	4	2	1	0
									S	Z	H	P/V	N	C
Jump	JR NZ, j	00 100 000 < j-2 >		REL	2 4	2 2	6 8	continue : Z = 1 PC <sub>R</sub> + j → PC <sub>R</sub> : Z = 0	●	●	●	●	●	●
Return	RET	11 001 001		IMP	3	1	9	(SP) <sub>M</sub> → PCL <sub>r</sub> (SP + 1) <sub>M</sub> → PCH <sub>r</sub> SP <sub>R</sub> + 2 → SP <sub>R</sub>	●	●	●	●	●	●
	RET f	11 f   000		IMP	3	1	5(f : false)	continue : f is false	●	●	●	●	●	●
					4	1	10(f : true)	RET : f is true						
	RETI	11 101 101 01 001 101		IMP	4	2	12	Return from interrupt	●	●	●	●	●	●
	RETN	11 101 101 01 000 101		IMP	4	2	12	Return from non-maskable interrupt	●	●	●	●	●	●
Restart	RST v	11 v   111		IMP	5	1	11	PCH <sub>r</sub> → (SP - 1) <sub>M</sub> PCL <sub>r</sub> → (SP - 2) <sub>M</sub> 0 → PCH <sub>r</sub> v → PCL <sub>r</sub> SP <sub>R</sub> - 2 → SP <sub>R</sub>	●	●	●	●	●	●
INPUT	IN A, (m)	11 011 011 < m >	IO	IMP	3	2	9	(Am) <sub>I</sub> → A <sub>r</sub> m → A <sub>0</sub> ~ A <sub>7</sub> A <sub>r</sub> → A <sub>8</sub> ~ A <sub>15</sub>	●	●	●	●	●	●

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
									7	6	4	2	1	0
			S	D					S	Z	H	P/V	N	C
INPUT	IN g, (C)	11 101 101 01 g 000	IO	REG	3	2	9	(BC) <sub>I</sub> → g <sub>r</sub> g = 110 : Only the flags will change. C <sub>r</sub> → A <sub>0</sub> ~ A <sub>7</sub> B <sub>r</sub> → A <sub>8</sub> ~ A <sub>15</sub>	↑	↑	R	P	R	●
	IN0 g, (m)	11 101 101 00 g 000 < m >	IO	REG	4	3	12	(00m) <sub>I</sub> → g <sub>r</sub> g = 110 : Only the flags will change. m → A <sub>0</sub> ~ A <sub>7</sub> 00 → A <sub>8</sub> ~ A <sub>15</sub>	↑	↑	R	P	R	●
	IND	11 101 101 10 101 010	IO	REGI	4	2	12	(BC) <sub>I</sub> → (HL) <sub>M</sub> HL <sub>R</sub> - 1 → HL <sub>R</sub> B <sub>r</sub> - 1 → B <sub>r</sub> C <sub>r</sub> → A <sub>0</sub> ~ A <sub>7</sub> B <sub>r</sub> → A <sub>8</sub> ~ A <sub>15</sub>	X	③ ↑	X	X	④ ↑	X
	INDR	11 101 101 10 111 010	IO	REGI	6 4	2 2	14 (B <sub>r</sub> ≠ 0) 12 (B <sub>r</sub> = 0)	<div style="display: inline-block; vertical-align: middle;">           (BC)<sub>I</sub> → (HL)<sub>M</sub>            HL<sub>R</sub> - 1 → HL<sub>R</sub>            B<sub>r</sub> - 1 → B<sub>r</sub> </div> Repeat Q until B <sub>r</sub> = 0 C <sub>r</sub> → A <sub>0</sub> ~ A <sub>7</sub> B <sub>r</sub> → A <sub>8</sub> ~ A <sub>15</sub>	X	S	X	X	④ ↑	X

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles		States	Operation	Flag					
			S	D					7	6	4	2	1	0
									S	Z	H	P/V	N	C
INPUT	INI	11 101 101 10 100 010	IO	REGI	4	2	12	(BC) <sub>I</sub> → (HL) <sub>M</sub> HL <sub>R</sub> + 1 → HL <sub>R</sub> B <sub>r</sub> - 1 → B <sub>r</sub> C <sub>r</sub> → A <sub>0</sub> ~ A <sub>7</sub> B <sub>r</sub> → A <sub>8</sub> ~ A <sub>15</sub>	③		④			
								X	↑	X	X	↑	X	
	INIR	11 101 101 10 110 010	IO	REGI	6	2	14 (B <sub>r</sub> ≠ 0)	(BC) <sub>I</sub> → (HL) <sub>M</sub> HL <sub>R</sub> + 1 → HL <sub>R</sub> B <sub>r</sub> - 1 → B <sub>r</sub>	X	S	X	X	④	X
								Q					↑	
					4	2	12 (B <sub>r</sub> = 0)	Repeat Q until B <sub>r</sub> = 0 C <sub>r</sub> → A <sub>0</sub> ~ A <sub>7</sub> B <sub>r</sub> → A <sub>8</sub> ~ A <sub>15</sub>						
OUTPUT	OUT (m), A	11 010 011 < m >	IMP	IO	4	2	10	A <sub>r</sub> → (Am) <sub>I</sub> m → A <sub>0</sub> ~ A <sub>7</sub> A <sub>r</sub> → A <sub>8</sub> ~ A <sub>15</sub>	●	●	●	●	●	●

③ Z = 1 : B<sub>r</sub> - 1 = 0    ④ N = 1 : MSB of Data = 1  
 Z = 0 : B<sub>r</sub> - 1 ≠ 0    N = 0 : MSB of Data = 0

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag										
			S	D					7	6	4	2	1	0					
									S	Z	H	P/V	N	C					
OUTPUT	OUT (C).g	11 101 101 01 g 001	REG	IO	4	2	10	$g_r \rightarrow (BC)_I$ $C_r \rightarrow A_0 \sim A_7$ $B_r \rightarrow A_8 \sim A_{15}$	●	●	●	●	●	●					
	OUT0 (m).g	11 101 101 00 g 001 < m >	REG	IO	5	3	13	$g_r \rightarrow (00m)_I$ $m \rightarrow A_0 \sim A_7$ $00 \rightarrow A_8 \sim A_{15}$	●	●	●	●	●	●					
	OTDM	11 101 101 10 001 011	REGI	IO	6	2	14	$(HL)_M \rightarrow (00C)_I$ $HL_R - 1 \rightarrow HL_R$ $C_r - 1 \rightarrow C_r$ $B_r - 1 \rightarrow B_r$ $C_r \rightarrow A_0 \sim A_7$ $00 \rightarrow A_8 \sim A_{15}$	③	④	↑	↑	P	↑	↑				
	OTDMR	11 101 101 10 011 011	REGI	IO	8 6	2 2	16 ( $B_r \neq 0$ ) 14 ( $B_r = 0$ )	$Q \left[ \begin{array}{l} (HL)_M \rightarrow (00C)_I \\ HL_R - 1 \rightarrow HL_R \\ C_r - 1 \rightarrow C_r \\ B_r - 1 \rightarrow B_r \end{array} \right.$ Repeat Q until $B_r = 0$ $C_r \rightarrow A_0 \sim A_7$ $00 \rightarrow A_8 \sim A_{15}$											
													④	R	S	R	S	↑	R

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
									7	6	4	2	1	0
			S	D					S	Z	H	P/V	N	C
OUTPUT	OTIM	11 101 101 10 000 011	REGI	IO	6	2	14	$(HL)_M \rightarrow (00C)_I$ $HL_R + 1 \rightarrow HL_R$ $C_r + 1 \rightarrow C_r$ $B_r - 1 \rightarrow B_r$ $C_r \rightarrow A_0 \sim A_7$ $00 \rightarrow A_8 \sim A_{15}$	<sup>③</sup> $\downarrow \downarrow \downarrow P \downarrow \downarrow$					
	OTIMR	11 101 101 10 010 011	REGI	IO	8  6	2  2	16 ( $B_r \neq 0$ ) 14 ( $B_r = 0$ )	$\left[ \begin{array}{l} (HL)_M \rightarrow (00C)_I \\ HL_R + 1 \rightarrow HL_R \\ C_r + 1 \rightarrow C_r \\ B_r - 1 \rightarrow B_r \end{array} \right. Q$  Repeat Q until $B_r = 0$ $C_r \rightarrow A_0 \sim A_7$ $00 \rightarrow A_8 \sim A_{15}$	<sup>④</sup> R S R S $\downarrow$ R					

Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
									7	6	4	2	1	0
			S	D					S	Z	H	P/V	N	C
OUTPUT	OUTD	11 101 101 10 101 011	REGI	IO	4	2	12	$(HL)_M \rightarrow (BC)_I$ $HL_R - 1 \rightarrow HL_R$ $B_r - 1 \rightarrow B_r$ $C_r \rightarrow A_0 \sim A_7$ $B_r \rightarrow A_8 \sim A_{15}$	<sup>③</sup> X $\uparrow$ X X $\uparrow$ X					
	OTDR	11 101 101 10 111 011	REGI	IO	6 4	2 2	14 12 (B <sub>r</sub> ≠ 0) (B <sub>r</sub> = 0)	$(HL)_M \rightarrow (BC)_I$ Q $HL_R - 1 \rightarrow HL_R$ $B_r - 1 \rightarrow B_r$ Repeat Q until B <sub>r</sub> = 0 $C_r \rightarrow A_0 \sim A_7$ $B_r \rightarrow A_8 \sim A_{15}$	<sup>④</sup> X S X X $\uparrow$ X					
	OUTI	11 101 101 10 100 011	REGI	IO	4	2	12	$(HL)_M \rightarrow (BC)_I$ $HL_R + 1 \rightarrow HL_R$ $B_r - 1 \rightarrow B_r$ $C_r \rightarrow A_0 \sim A_7$ $B_r \rightarrow A_8 \sim A_{15}$	<sup>③</sup> X $\uparrow$ X X $\uparrow$ X					



Operation name	MNEMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag					
									7	6	4	2	1	0
			S	D					S	Z	H	P/V	N	C
OUTPUT	OTIR	11 101 101 10 110 011	REGI	IO	6  4	2  2	14 ( $B_r \neq 0$ ) 12 ( $B_r = 0$ )	$(HL)_M \rightarrow (BC)_I$ $Q \left[ \begin{array}{l} HL_R + 1 \rightarrow HL_R \\ B_r - 1 \rightarrow B_r \end{array} \right.$ Repeat Q until $B_r = 0$ $C_r \rightarrow A_0 \sim A_7$ $B_r \rightarrow A_8 \sim A_{15}$	X	S	X	X	<sup>④</sup> ↑	X
	TSTIO m	11 101 101 01 110 100  < m >	IMMED/ IO		4	3	12	$(00C)_I \cdot m$ $C_r \rightarrow A_0 \sim A_7$ $00 \rightarrow A_8 \sim A_{15}$	↑	↑	S	P	R	R

③  $Z = 1 : B_r - 1 = 0, Z = 0 : B_r - 1 \neq 0$

④  $N = 1 : \text{MSB of Data} = 1, N = 0 : \text{MSB of Data} = 0$

Operation name	MENMONICS	OP-code	Addressing Modes		Machine Cycles	Bytes	States	Operation	Flag
									7 6 4 2 1 0
			S	D					S Z H P/V N C
Special Function	DAA	00 100 111	IMP	IMP	2	1	4	Decimal Adjust Accumulator	↑ ↑ ↑ P ● ↑
Carry Control	CCF	00 111 111			1	1	3	$\bar{c} \rightarrow c$	● ● R ● R ↑
	SCF	00 110 111			1	1	3	$1 \rightarrow c$	● ● R ● R S
CPU Control	DI	11 110 011			1	1	3	$0 \rightarrow IEF_1, IEF_2$ ⑤	● ● ● ● ● ●
	EI	11 111 011			1	1	3	$1 \rightarrow IEF_1, IEF_2$ ⑤	● ● ● ● ● ●
	HALT	01 110 110			1	1	3	CPU halted	● ● ● ● ● ●
	IM 0	11 101 101			2	2	6	Interrupt mode 0	● ● ● ● ● ●
		01 000 110							
	IM 1	11 101 101			2	2	6	Interrupt mode 1	● ● ● ● ● ●
		01 010 110							
	IM 2	11 101 101			2	2	6	Interrupt mode 2	● ● ● ● ● ●
		01 011 110							
	NOP	00 000 000			1	1	3	No operation	● ● ● ● ● ●
	SLP	11 101 101			2	2	8	Sleep	● ● ● ● ● ●
		01 110 110							

⑤ DIまたはEIの最後では割込みはサンプリングされません。

# 各バスサイクルの状態一覧表

\* : 不定 Z : ハイインピーダンス

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
ADD HL, ww	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub> ~MC <sub>5</sub>	T <sub>1</sub> T <sub>1</sub> T <sub>1</sub> T <sub>1</sub>	*	Z	1	1	1	1	1	1	1
ADD IX, xx ADD IY, yy	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub> ~MC <sub>6</sub>	T <sub>1</sub> T <sub>1</sub> T <sub>1</sub> T <sub>1</sub>	*	Z	1	1	1	1	1	1	1
ADC HL, ww SBC HL, ww	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub> ~MC <sub>6</sub>	T <sub>1</sub> T <sub>1</sub> T <sub>1</sub> T <sub>1</sub>	*	Z	1	1	1	1	1	1	1
ADD A, g ADC A, g SUB g	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	HALT	ST
SBC A, g AND g OR g XOR g CP g	MC <sub>2</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
ADD A, m ADC A, m SUB m SBC A, m AND m OR m XOR m CP m	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	m	0	1	0	1	1	1	1
ADD A, (HL) ADC A, (HL) SUB (HL) SBC A, (HL) AND (HL) OR (HL) XOR (HL) CP (HL)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	0	1	0	1	1	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
ADD A, (IX + d) ADD A, (IY + d) ADC A, (IX + d) ADC A, (IY + d) SUB (IX + d) SUB (IY + d) SBC A, (IX + d) SBC A, (IY + d) AND (IX + d) AND (IY + d) OR (IX + d) OR (IY + d) XOR (IX + d) XOR (IY + d) CP (IX + d) CP (IY + d)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	d	0	1	0	1	1	1	1
	MC <sub>4</sub> ~MC <sub>5</sub>	T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX + d IY + d	DATA	0	1	0	1	1	1	1
BIT b, g	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
BIT b,(HL)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	0	1	0	1	1	1	1
BIT b,(IX + d) BIT b,(IY + d)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	d	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	3rd op-code Address	3rd op-code	0	1	0	1	0	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX + d IY + d	DATA	0	1	0	1	1	1	1
CALL mn	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
CALL mn	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-1	PCH	1	0	0	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-2	PCL	1	0	0	1	1	1	1
CALL f, mn (If condition is false)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
CALL f, mn (If condition is true)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-1	PCH	1	0	0	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-2	PCL	1	0	0	1	1	1	1

Instruction	Machine Cycle		ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
CCF	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
CPI CPD	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	0	1	0	1	1	1	1
	MC <sub>4</sub> ~MC <sub>6</sub>	T <sub>1</sub> T <sub>1</sub> T <sub>1</sub>	*	Z	1	1	1	1	1	1	1
CPIR CPDR (If BC <sub>R</sub> ≠ 0 and A <sub>r</sub> ≠ (HL) <sub>M</sub> )	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	0	1	0	1	1	1	1
	MC <sub>4</sub> ~MC <sub>8</sub>	T <sub>1</sub> T <sub>1</sub> T <sub>1</sub> T <sub>1</sub> T <sub>1</sub>	*	Z	1	1	1	1	1	1	1
CPIR CPDR (If BC <sub>R</sub> = 0 or A <sub>r</sub> = (HL) <sub>M</sub> )	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0



Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
CPIR CPDR (If $BC_R = 0$ or $A_r = (HL)_M$ )	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	0	1	0	1	1	1	1
	MC <sub>4</sub> ~MC <sub>6</sub>	Ti Ti Ti	*	Z	1	1	1	1	1	1	1
CPL	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
DAA	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	Ti	*	Z	1	1	1	1	1	1	1
DI	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
DJNZ j (If $B_r \neq 0$ )	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	Ti * 1	*	Z	1	1	1	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	j-2	0	1	0	1	1	1	1
	MC <sub>4</sub> ~MC <sub>5</sub>	Ti Ti	*	Z	1	1	1	1	1	1	1

\* 1 このステートの直後で、DMA、リフレッシュ、BUS、RELEASEを実行することはできません。(要求を入力しても無視されます。)

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
DJNZ j (If $B_r = 0$ )	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>i</sub> * 1	*	Z	1	1	1	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	j-2	0	1	0	1	1	1	1
EI	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
EX DE, HL EXX	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
EX AF, AF <sup>*</sup>	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
EX (SP), HL	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP	DATA	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP + 1	DATA	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP + 1	H	1	0	0	1	1	1	1

\*1 このステートの直後で、DMA、リフレッシュ、BUS、RELEASEを実行することはできません。(要求を入力しても無視されます。)

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
EX (SP).HL	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP	L	1	0	0	1	1	1	1
EX (SP), IX EX (SP), IY	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP	DATA	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP + 1	DATA	0	1	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP + 1	IXH IYH	1	0	0	1	1	1	1
	MC <sub>7</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP	IXL IYL	1	0	0	1	1	1	1
HALT	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	—	—	Next op-code Address	Next op-code	0	1	0	1	0	0	0
IM 0 IM 1 IM 2	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{\text{RD}}$	$\overline{\text{WR}}$	$\overline{\text{ME}}$	$\overline{\text{IOE}}$	$\overline{\text{LIR}}$	$\overline{\text{HALT}}$	ST
INC g DEC g	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
INC (HL) DEC (HL)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	1	0	0	1	1	1	1
INC (IX + d) INC (IY + d) DEC (IX + d) DEC (IY + d)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	d	0	1	0	1	1	1	1
	MC <sub>4</sub> ~MC <sub>5</sub>	T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX + d IY + d	DATA	0	1	0	1	1	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
INC (IX + d) INC (IY + d) DEC (IX + d) DEC (IY + d)	MC <sub>7</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>8</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX + d IY + d	DATA	1	0	0	1	1	1	1
INC ww DEC ww	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
INC IX INC IY DEC IX DEC IY	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
IN A, (m)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	m	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	m to A <sub>0</sub> ~ A <sub>7</sub> A to A <sub>8</sub> ~ A <sub>15</sub>	DATA	0	1	1	0	1	1	1
IN g, (C)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
IN g. (C)	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	BC	DATA	0	1	1	0	1	1	1
IN0 g.(m)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	m	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	m to A <sub>0</sub> ~ A <sub>7</sub> 00H to A <sub>8</sub> ~ A <sub>15</sub>	DATA	0	1	1	0	1	1	1
INI IND	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	BC	DATA	0	1	1	0	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	1	0	0	1	1	1	1
INIR INDR (If B <sub>r</sub> ≠ 0)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
INIR INDR (If $B_r \neq 0$ )	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	BC	DATA	0	1	1	0	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	1	0	0	1	1	1	1
	MC <sub>5</sub> ~MC <sub>6</sub>	T <sub>1</sub> T <sub>1</sub>	*	Z	1	1	1	1	1	1	1
INIR INDR (If $B_r = 0$ )	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	BC	DATA	0	1	1	0	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	1	0	0	1	1	1	1
JP mn	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
JP f, mn (If f is false)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
JP f, mn (If f is true)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
JP (HL)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
JP (IX) JP (IY)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
JR j	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	j-2	0	1	0	1	1	1	1
	MC <sub>3</sub> ~MC <sub>4</sub>	T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1



Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
JR C, j JR NC, j JR Z, j JR NZ, j (If condition is false)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	j-2	0	1	0	1	1	1	1
JR C, j JR NC, j JR Z, j JR NZ, j (If condition is true)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	j-2	0	1	0	1	1	1	1
	MC <sub>3</sub>										
	~MC <sub>4</sub>	T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
LD g, g'	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
LD g, m	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	m	0	1	0	1	1	1	1
LD g, (HL)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	0	1	0	1	1	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
LD g, (IX + d) LD g, (IY + d)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	d	0	1	0	1	1	1	1
	MC <sub>4</sub> ~MC <sub>5</sub>	T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX + d IY + d	DATA	0	1	0	1	1	1	1
LD (HL), g	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	g	1	0	0	1	1	1	1
LD (IX + d), g LD (IY + d), g	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	d	0	1	0	1	1	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
LD (IX+d), g LD (IY+d), g	MC <sub>4</sub> ~MC <sub>6</sub>	T <sub>i</sub> T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>7</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX+d IY+d	g	1	0	0	1	1	1	1
LD (HL), m	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	m	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	1	0	0	1	1	1	1
LD (IX+d), m LD (IY+d), m	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	d	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX+d IY+d	DATA	1	0	0	1	1	1	1
LD A, (BC) LD A, (DE)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
LD A, (BC) LD A, (DE)	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	BC DE	DATA	0	1	0	1	1	1	1
LD A, (mn)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn	DATA	0	1	0	1	1	1	1
LD (BC), A LD (DE), A	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	BC DE	A	1	0	0	1	1	1	1
LD (mn), A	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn	A	1	0	0	1	1	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
LD A, I LD A, R LD I, A LD R, A	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
LD ww, mn	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
LD IX, mn LD IY, mn	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
LD HL, (mn)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
LD HL, (mn)	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn	DATA	0	1	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn + 1	DATA	0	1	0	1	1	1	1
LD ww, (mn)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn	DATA	0	1	0	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn + 1	DATA	0	1	0	1	1	1	1
LD IX, (mn) LD IY, (mn)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
LD IX, (mn) LD IY, (mn)	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn	DATA	0	1	0	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn + 1	DATA	0	1	0	1	1	1	1
LD (mn), HL	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn	L	1	0	0	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn + 1	H	1	0	0	1	1	1	1
LD (mn), ww	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
LD (mn), ww	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>5</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn	wwL	1	0	0	1	1	1	1
	MC <sub>7</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn + 1	wwH	1	0	0	1	1	1	1
LD (mn), IX LD (mn), IY	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	n	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd operand Address	m	0	1	0	1	1	1	1
	MC <sub>5</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1



Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
LD (mn), IX LD (mn), IY	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn	IXL IYL	1	0	0	1	1	1	1
	MC <sub>7</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	mn + 1	IXH IYH	1	0	0	1	1	1	1
LD SP, HL	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
LD SP, IX LD SP, IY	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
LDI LDD	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	DE	DATA	1	0	0	1	1	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
LDIR LDDR (If $BC_R \neq 0$ )	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	DE	DATA	1	0	0	1	1	1	1
	MC <sub>5</sub> ~MC <sub>6</sub>	T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
LDIR LDDR (If $BC_R = 0$ )	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	DE	DATA	1	0	0	1	1	1	1
MLT ww	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
MLT ww	MC <sub>3</sub> ~MC <sub>13</sub>	T <sub>i</sub> T <sub>i</sub> T <sub>i</sub> T <sub>i</sub> T <sub>i</sub> T <sub>i</sub> T <sub>i</sub> T <sub>i</sub> T <sub>i</sub> T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
NEG	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
NOP	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
OUT (m), A	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	m	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	m to A <sub>0</sub> ~A <sub>7</sub> A to A <sub>8</sub> ~A <sub>15</sub>	A	1	0	1	0	1	1	1
OUT (C), g	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	BC	g	1	0	1	0	1	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
OUT0 (m), g	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	m	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	m to A <sub>0</sub> ~ A <sub>7</sub> 00H to A <sub>8</sub> ~ A <sub>15</sub>	g	1	0	1	0	1	1	1
OTIM OTDM	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	0	1	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	C to A <sub>0</sub> ~ A <sub>7</sub> 00H to A <sub>8</sub> ~ A <sub>15</sub>	DATA	1	0	1	0	1	1	1
	MC <sub>6</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
OTIMR OTDMR (If B <sub>r</sub> ≠ 0)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
OTIMR OTDMR (If $B_r \neq 0$ )	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	0	1	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	C to A <sub>0</sub> ~ A <sub>7</sub> 00H to A <sub>8</sub> ~ A <sub>15</sub>	DATA	1	0	1	0	1	1	1
	MC <sub>6</sub> ~ MC <sub>8</sub>	T <sub>i</sub> T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
OTIMR OTDMR (If $B_r = 0$ )	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	0	1	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	C to A <sub>0</sub> ~ A <sub>7</sub> 00H to A <sub>8</sub> ~ A <sub>15</sub>	DATA	1	0	1	0	1	1	1
	MC <sub>6</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
OUTI OUTD	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
OUTI OUTD	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	BC	DATA	1	0	1	0	1	1	1
OTIR OTDR (If B <sub>r</sub> ≠ 0)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	BC	DATA	1	0	1	0	1	1	1
	MC <sub>5</sub> ~MC <sub>6</sub>	T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
OTIR OTDR (If B <sub>r</sub> = 0)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	BC	DATA	1	0	1	0	1	1	1
POP zz	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP	DATA	0	1	0	1	1	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
POP zz	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP + 1	DATA	0	1	0	1	1	1	1
POP IX POP IY	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP	DATA	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP + 1	DATA	0	1	0	1	1	1	1
PUSH zz	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub> ~MC <sub>3</sub>	T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-1	zzH	1	0	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-2	zzL	1	0	0	1	1	1	1
PUSH IX PUSH IY	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub> ~MC <sub>4</sub>	T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-1	IXH IYH	1	0	0	1	1	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
PUSH IX PUSH IY	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-2	IXL IYL	1	0	0	1	1	1	1
RET	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP	DATA	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP + 1	DATA	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP + 1	DATA	0	1	0	1	1	1	1
RET f (If condition is false)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>									
	~MC <sub>3</sub>	T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
RET f (If condition is true)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP	DATA	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP + 1	DATA	0	1	0	1	1	1	1
RETI RETN	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0



Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
RETI RETN	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP	DATA	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP + 1	DATA	0	1	0	1	1	1	1
RLCA RLA RRCA RRA	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
RLC g RL g RRC g RR g SLA g SRA g SRL g	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
RLC (HL) RL (HL) RRC (HL) RR (HL) SLA (HL)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	0	1	0	1	1	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
SRA (HL) SRL (HL)	MC <sub>4</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	1	0	0	1	1	1	1
RLC (IX+d) RLC (IY+d)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
RL (IX+d) RL (IY+d)	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
RRC (IX+d) RRC (IY+d)	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	d	0	1	0	1	1	1	1
RR (IX+d) RR (IY+d)	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	3rd op-code Address	3rd op-code	0	1	0	1	0	1	1
SLA (IX+d) SLA (IY+d)	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX+d IY+d	DATA	0	1	0	1	1	1	1
SRA (IX+d) SRA (IY+d)	MC <sub>6</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
SRL (IX+d) SRL (IY+d)	MC <sub>7</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX+d IY+d	DATA	1	0	0	1	1	1	1
RLD RRD	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
RLD RRD											
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	0	1	0	1	1	1	1
	MC <sub>4</sub> ~MC <sub>7</sub>	T <sub>i</sub> T <sub>i</sub> T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>8</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	1	0	0	1	1	1	1
RST v											
	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub> ~MC <sub>3</sub>	T <sub>i</sub> T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP - 1	PCH	1	0	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP - 2	PCL	1	0	0	1	1	1	1
SCF	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
SET b, g RES b, g											
	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1

Instruction	Machine, Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
SET b,(HL) RES b,(HL)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	1	0	0	1	1	1	1
SET b,(IX+d) SET b,(IY+d) RES b,(IX+d) RES b,(IY+d)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	d	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	3rd op-code Address	3rd op-code	0	1	0	1	0	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX+d IY+d	DATA	0	1	0	1	1	1	1
	MC <sub>6</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
	MC <sub>7</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	IX+d IY+d	DATA	1	0	0	1	1	1	1
SLP	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	—	—	7FFFFH	Z	1	1	1	1	1	0	1
TSTIO m	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	m	0	1	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	C to A <sub>0</sub> ~A <sub>7</sub> 00H to A <sub>8</sub> ~A <sub>15</sub>	DATA	0	1	1	0	1	1	1
TST g	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub>	*	Z	1	1	1	1	1	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
TST m	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st operand Address	m	0	1	0	1	1	1	1
TST (HL)	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	1st op-code Address	1st op-code	0	1	0	1	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	2nd op-code Address	2nd op-code	0	1	0	1	0	1	1
	MC <sub>3</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	HL	DATA	0	1	0	1	1	1	1

# INTERRUPT

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
$\overline{NMI}$	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	Next op-code Address (PC)	—	0	1	0	1	0	1	0
	MC <sub>2</sub> ~MC <sub>3</sub>	T <sub>1</sub> T <sub>1</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP - 1	PCH	1	0	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP - 2	PCL	1	0	0	1	1	1	1
	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>w</sub> T <sub>w</sub> T <sub>3</sub>	Next op-code Address (PC)	1st op-code	1	1	1	0	0	1	0
$\overline{INT_0}$ MODE 0 (RST INSERTED)	MC <sub>2</sub> ~MC <sub>3</sub>	T <sub>1</sub> T <sub>1</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP - 1	PCH	1	0	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP - 2	PCL	1	0	0	1	1	1	1
	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>w</sub> T <sub>w</sub> T <sub>3</sub>	Next op-code Address (PC)	1st op-code	1	1	1	0	0	1	0
$\overline{INT_0}$ MODE 0 (CALL INSERTED)	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	PC	n	0	1	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	PC + 1	m	0	1	0	1	1	1	1

Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
$\overline{INT}_0$ MODE 0 (CALL INSERTED)	MC <sub>1</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP - 1	PC + 2 (H)	1	0	0	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP - 2	PC + 2 (L)	1	0	0	1	1	1	1
$\overline{INT}_0$ MODE 1	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>w</sub> T <sub>w</sub> T <sub>3</sub>	Next op-code Address (PC)	Z	1	1	1	0	0	1	0
	MC <sub>2</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP - 1	PCH	1	0	0	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP - 2	PCL	1	0	0	1	1	1	1
$\overline{INT}_0$ MODE 2	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>w</sub> T <sub>w</sub> T <sub>3</sub>	Next op-code Address (PC)	ベクタ	1	1	1	0	0	1	0
	MC <sub>2</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP - 1	PCH	1	0	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP - 2	PCL	1	0	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	I, ベクタ	DATA	0	1	0	1	1	1	1



Instruction	Machine Cycle	States	ADDRESS	DATA	$\overline{RD}$	$\overline{WR}$	$\overline{ME}$	$\overline{IOE}$	$\overline{LIR}$	$\overline{HALT}$	ST
$\overline{INT}_0$ MODE 2	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	I, ベクタ+1	DATA	0	1	0	1	1	1	1
$\overline{INT}_1$ $\overline{INT}_2$ 内部割込み	MC <sub>1</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>w</sub> T <sub>w</sub> T <sub>3</sub>	Next op-code Address (PC)	Z	1	1	1	1	1	1	0
	MC <sub>2</sub>	T <sub>i</sub>	*	Z	1	1	1	1	1	1	1
	MC <sub>3</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-1	PCH	1	0	0	1	1	1	1
	MC <sub>4</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	SP-2	PCL	1	0	0	1	1	1	1
	MC <sub>5</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	I, ベクタ	DATA	0	1	0	1	1	1	1
	MC <sub>6</sub>	T <sub>1</sub> T <sub>2</sub> T <sub>3</sub>	I, ベクタ+1	DATA	0	1	0	1	1	1	1

表-1 HD64180オペコードマップ

第1オペコード XXの形の命令

		LO		HI		ww (LO=ALL)				g (LO=0~7)				LO=0~7																
						BC	DE	HL	SP					BC	DE	HL	AF	zz												
						g (LO=0~7)								NZ	NC	PO	P	f												
						B	D	H	(HL)	B	D	H	(HL)	00H	10H	20H	30H	v												
						0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111									
						0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F									
S (HI=ALL)	B	0000	0	NOP	DJNZ	JR NZ	JR NC	LD g, s				ADD A SUB s AND s OR s				RET f				0										
	C	0001	1	LD ww, mn												注1)				POP zz				1						
	D	0010	2	LD (ww), A		LD (mn), HL	LD (mn), A									JP f, mn				2										
				JP mn												OUT (m)	EX (SP)	DI	3											
	E	0011	3	INC ww												CALL f, mn				4										
	H	0100	4	INC g												PUSH zz				5										
	L	0101	5	DEC g												RST v				6										
	(HL)	0110	6	LD g, m				注2)				HALT	注2)	注2)	注2)	注2)	ADD A m	SUB m	AND m	OR m	6									
	A	0111	7	RLCA	RLA	DAA	SCF	LD g, s				ADC A SBC A XOR s CP s				RET f				7										
	B	1000	8	EXAF, AF	JR	JR Z	JR C									JP f, mn				A										
	C	1001	9	ADD HL, ww												RET	EXX	JP (HL)	LD SP, HL	9										
	D	1010	A	LD A, (ww)		LD HL, (mn)	LD A, (mn)									表2 IN A, (m) EXDE, HL EI				B										
	E	1011	B	DEC ww												CALL f, mn				C										
	H	1100	C	INC g												CALL mn	注3)	表3	注3)	D										
	L	1101	D	DEC g												ADC, A m	SBC A m	XOR m	CP m	E										
	(HL)	1110	E	LD g, m				注2)				注2)	注2)	注2)	注2)	RST v				F										
	A	1111	F	RRCA	RRA	CPL	CCF	0	1	2	3	4	5	6	7	C	D	E	F											
				C	E	L	A	C	E	L	A	8	9	A	B	Z	C	PE	M	f										
				g (LO=8~F)												08H	18H	28H	38H	v										
																LO=8~F														

- 注1) gが(HL)に置きかわる。  
注2) sが(HL)に置きかわる。  
注3) オペコードの先頭にDDを付加すると(DD XX)、オペランドとして、HL, (HL)をもつ命令のみ HL→IX と置きかえられて同じオペレーションをする。  
(HL)→(IX+d)

(例) 22H; LD (mn), HL



DDH 22H; LD (mn), IX

同様にオペコードの先頭にFDを付加すると、(FD XX)として HL→IY  
(HL)→(IY+d)

と置きかえられて、同じオペレーションをする。

(例) 34H; INC (HL)



FDH 34H; INC (IY+d)

ただし、例外として、E9HのJP(HL)は、先頭にDDH, FDHを付加した場合、(HL)は(IX), (IY)に置きかわられる。

また、EBHのEX DE, HL は、先頭にDDH, FDHを付加しても、HLの置き換えは無く未定義命令となる。

表-2 HD64180オペコードマップ

第2オペコード CBXXの形の命令

		HI	b (L0=0~7)																
			0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111	
			0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
g (HI=ALL)	B	0000	0																0
	C	0001	1																1
	D	0010	2																2
	E	0011	3																3
	H	0100	4	RLO g	RL g	SLA g	BIT b,g				RES b,g				SET b,g				4
	L	0101	5				注1)				注1)				注1)				5
	(HL)	0110	6	注1)	注1)	注1)	注1)				注1)				注1)				6
	A	0111	7				注1)				注1)				注1)				7
	B	1000	8																8
	C	1001	9																9
	D	1010	A																A
	E	1011	B																B
	H	1100	C	RRC g	RR g	SRA g	SRL g	BIT b,g				RES b,g				SET b,g			C
	L	1101	D					注1)				注1)				注1)			D
	(HL)	1110	E	注1)	注1)	注1)	注1)	注1)				注1)				注1)			E
	A	1111	F					注1)				注1)				注1)			F
			0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
							1	3	5	7	1	3	5	7	1	3	5	7	
			b (L0=8~F)																

注1) オペコードの先頭にDDHを付加し、DD CB d XXというオペコードで(HL)を(IX+d)で置きかえた命令を実行する。  
 同様にオペコードの先頭にFDHを付加し、FD CB d XXというオペコードで(HL)を(IY+d)で置きかえた命令を実行する。

表-3 HD64180オペコードマップ

第2オペコード EDXXの形の命令

LO \ HI		ww (LO=ALL)																	
		BC				DE				HL				SP					
		g (LO=0~7)																	
		B	D	H		B	D	H											
LO	HI	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111		
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F		
0000	0	IN0 g, (m)				IN g, (C)								LDI	LDIR			0	
0001	1	OUT0 (m),g				OUT (C),g								CPI	CPIR			1	
0010	2					SBC HL,ww								INI	INIR			2	
0011	3					LD(mn), ww				OTIM		OTIMR	OUTI	OTIR				3	
0100	4	TST g			TST(HL)	NEG			TST m	TST0 m								4	
0101	5					RETN												5	
0110	6					IM0	IM1	SLP									6		
0111	7					LD I,A	LD A,I	RRD											7
1000	8	IN0 g, (m)				IN g, (C)								LDD	LDDR			8	
1001	9	OUT0 (m),g				OUT (C),g								CPD	CPDR			9	
1010	A					ADC HL,ww								IND	INDR			A	
1011	B					LD ww, (mn)				OTDM		OTDMR	OUTD	OTDR			B		
1100	C	TST g				MLT ww												C	
1101	D					RETI												D	
1110	E					IM2												E	
1111	F					LD R,A				LD A,R	RLD							F	
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F		
		C	E	L	A	C	E	L	A										
		g (LO=8~F)																	

## 内蔵 I/Oレジスタ早見表

I/Oレジスタのアドレスは、上位8ビットすべて“0”であり、下位8ビット中MSBから2ビットは、I/Oコントロールレジスタ内のIOA7とIOA6により設定できます。以下の表に示すアドレスは、IOA7とIOA6が“0”の場合です。

レジスタ	ニーモニック	アドレス	備 考
ASCIコントロールレジスタAチャンネル0 : CNTLA0           ASCIコントロールレジスタAチャンネル1 : CNTLA1		00	ビット
			リセット時
			R/W
		01	ビット
			リセット時
			R/W
			MOD2, 1, 0
			000 : Start + 7bit Data + 1 stop
			001 : Start + 7bit Data + 2 stop
			010 : Start + 7bit Data + Parity + 1 stop
			011 : Start + 7bit Data + Parity + 2 stop
			100 : Start + 8bit Data + 1 stop
			101 : Start + 8bit Data + 2 stop
			110 : Start + 8bit Data + Parity + 1 stop
			111 : Start + 8bit Data + Parity + 2 stop

レジスタ	ニーモニク	アドレス	備 考								
ASCIコントロールレジスタBチャンネル0 :CNTLB0		0 2	ビット	MPBT	MP	CTS/PS	PE0	DR	SS2	SS1	SS0
			リセット時	不定	0	*	0	0	1	1	1
			R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
		* CTS : CTS端子の状態を取り込みます。 PS : クリアされた状態になっています。									
		0 3	ビット	MPBT	MP	CTS/PS	PE0	DR	SS2	SS1	SS0
			リセット時	不定	0	0	0	0	1	1	1
			R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
			総合分周比		PS=0 (分周比=10)		PS=1 (分周比=30)				
			SS2, 1, 0	DR=0(×16)	DR=1(×64)	DR=0(×16)	DR=1(×64)				
			0 0 0	φ÷ 160	φ÷ 640	φ÷ 480	φ÷ 1,920				
0 0 1	÷ 320		÷ 1,280	÷ 960	÷ 3,840						
ASCIコントロールレジスタBチャンネル1 :CNTLB1		0 3	0 1 0	÷ 640	÷ 2,560	÷ 1,920	÷ 7,680				
			0 1 1	÷ 1,280	÷ 5,120	÷ 3,840	÷ 15,360				
			1 0 0	÷ 2,560	÷ 10,240	÷ 7,680	÷ 30,720				
			1 0 1	÷ 5,120	÷ 20,480	÷ 15,360	÷ 61,440				
			1 1 0	÷ 10,240	÷ 40,960	÷ 30,720	÷ 122,880				
			1 1 1	外部クロック入力(÷40以上)							
			総合分周比		PS=0 (分周比=10)		PS=1 (分周比=30)				
		SS2, 1, 0	DR=0(×16)	DR=1(×64)	DR=0(×16)	DR=1(×64)					

レジスタ	ニーモニック	アドレス	備 考																														
ASCIIステータスレジスタチャンネル0 : STAT0	0 4	ビット	RDRF	OV RN	PE	FE	RIE	$\overline{DCD0}$	TDRE	TIE																							
		リセット時	0	0	0	0	0	*	**	0																							
		R/W	R	R	R	R	R/W	R	R	R/W																							
		* $\overline{DCD0}$ 端子の状態を取込みます。																															
			<table><tr><td colspan="7">** <math>\overline{CTS0}</math> 端子</td><td>TDRE</td></tr><tr><td colspan="7">L</td><td>1</td></tr><tr><td colspan="7">H</td><td>0</td></tr></table>							** $\overline{CTS0}$ 端子							TDRE	L							1	H							0
** $\overline{CTS0}$ 端子							TDRE																										
L							1																										
H							0																										
ASCIIモード/ステータスレジスタチャンネル1 : STAT1	0 5	ビット	RDRF	OV RN	PE	FE	RIE	CTS1E	TDRE	TIE																							
		リセット時	0	0	0	0	0	0	1	0																							
		R/W	R	R	R	R	R/W	R/W	R	R/W																							
ASCITランスミットデータレジスタチャンネル0 : TDR0	0 6																																
ASCITランスミットデータレジスタチャンネル1 : TDR1	0 7																																
ASCIIレシーブデータレジスタチャンネル0 : TSR0	0 8																																
ASCIIレシーブデータレジスタチャンネル1 : TSR1	0 9																																



レジスタ	ニーモニック	アドレス	備 考								
CSI／0コントロールレジスタ : CNTR	0 A	ビット	EF	EIE	RE	TE	—	SS2	SS1	SS0	
		リセット時	0	0	0	0	1	1	1	1	
		R／W	R	R／W	R／W	R／W		R／W	R／W	R／W	
		SS2, 1, 0	ボーレート		SS2, 1, 0	ボーレート					
		0 0 0	φ ÷ 20		1 0 0	φ ÷ 320					
		0 0 1	÷ 40		1 0 1	÷ 640					
		0 1 0	÷ 80		1 1 0	÷ 1,280					
		0 1 1	÷ 160		1 1 1	外部( ÷ 20以上)					
		CSI／0トランスミット／レシーブデータレジスタ : TRDR	0 B								
		タイマデータレジスタチャネル0L : TMDR0L	0 C								
タイマデータレジスタチャネル0H : TMDR0H	0 D										
タイマリロードレジスタチャネル0L : RLDR0L	0 E										
タイマリロードレジスタチャネル0H : RLDR0H	0 F										

レジスタ	ニーモニク	アドレス	備 考								
タイマコントロールレジスタ :TCR		1 0	ビット	TIF1	TIF0	TIE1	TIE0	TOC1	TOC0	TDE1	TDE0
			リセット時	0	0	0	0	0	0	0	0
			R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W
			TOC1,0	A18/Tout端子							
			0 0	タイマ出力禁止							
			0 1	トグル出力							
			1 0	"0"出力							
タイマデータレジスタチャンネル1L :TMDR1L		1 4	1 1	"1"出力							
タイマデータレジスタチャンネル1H :TMDR1H		1 5									
タイマリロードレジスタチャンネル1L :RLDR1L		1 6									
タイマリロードレジスタチャンネル1H :RLDR1H		1 7									
フリーランニングカウンタ :FRC		1 8	リードのみ								
DMAソースアドレスレジスタチャンネル0L :SAR0L		2 0									
DMAソースアドレスレジスタチャンネル0H :SAR0H		2 1									

レジスタ	ニーモニック	アドレス	備考										
DMAソースアドレスレジスタチャンネル0B : SAR0B		2 2	ビット0,1,2のみ使用 <table><tr><th>A<sub>18</sub>, A<sub>17</sub>, A<sub>16</sub></th><th></th></tr><tr><td>× 0 0</td><td>DREQ<sub>0</sub> (外部)</td></tr><tr><td>× 0 1</td><td>RDR0 (ASCI0)</td></tr><tr><td>× 1 0</td><td>RDR1 (ASCI1)</td></tr><tr><td>× 1 1</td><td>Not Used</td></tr></table>	A <sub>18</sub> , A <sub>17</sub> , A <sub>16</sub>		× 0 0	DREQ <sub>0</sub> (外部)	× 0 1	RDR0 (ASCI0)	× 1 0	RDR1 (ASCI1)	× 1 1	Not Used
A <sub>18</sub> , A <sub>17</sub> , A <sub>16</sub>													
× 0 0	DREQ <sub>0</sub> (外部)												
× 0 1	RDR0 (ASCI0)												
× 1 0	RDR1 (ASCI1)												
× 1 1	Not Used												
DMAデスティネーションアドレスレジスタチャンネル0L : DAR0L		2 3											
DMAデスティネーションアドレスレジスタチャンネル0H : DAR0H		2 4											
DMAデスティネーションアドレスレジスタチャンネル0B : DAR0B		2 5	ビット0,1,2のみ使用 <table><tr><th>A<sub>18</sub>, A<sub>17</sub>, A<sub>16</sub></th><th></th></tr><tr><td>× 0 0</td><td>DREQ<sub>0</sub> (外部)</td></tr><tr><td>× 0 1</td><td>TDR0 (ASCI0)</td></tr><tr><td>× 1 0</td><td>TDR1 (ASCI1)</td></tr><tr><td>× 1 1</td><td>Not Used</td></tr></table>	A <sub>18</sub> , A <sub>17</sub> , A <sub>16</sub>		× 0 0	DREQ <sub>0</sub> (外部)	× 0 1	TDR0 (ASCI0)	× 1 0	TDR1 (ASCI1)	× 1 1	Not Used
A <sub>18</sub> , A <sub>17</sub> , A <sub>16</sub>													
× 0 0	DREQ <sub>0</sub> (外部)												
× 0 1	TDR0 (ASCI0)												
× 1 0	TDR1 (ASCI1)												
× 1 1	Not Used												
DMAバイトカウントレジスタチャンネル0L : BCR0L		2 6											
DMAバイトカウントレジスタチャンネル0H : BCR0H		2 7											

レジスタ	ニーモニック	アドレス	備考																									
DMAメモリアドレスレジスタチャンネル1L : MAR1L		2 8	ビット0, 1, 2のみ使用																									
DMAメモリアドレスレジスタチャンネル1H : MAR1H		2 9																										
DMAメモリアドレスレジスタチャンネル1B : MAR1B		2 A																										
DMA I/Oアドレスレジスタチャンネル1L : IAR1L		2 B																										
DMA I/Oアドレスレジスタチャンネル1H : IAR1H		2 C																										
DMAバイトカウントレジスタチャンネル1L : BCR1L		2 E																										
DMAバイトカウントレジスタチャンネル1H : BCR1H		2 F																										
DMAステータスレジスタ : DSTAT		3 0	ビット リセット時 R/W	<table><tr><td>DE1</td><td>DE0</td><td>DWE1</td><td>DWE0</td><td>DIE1</td><td>DIE0</td><td>—</td><td>DME</td></tr><tr><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td></tr><tr><td>R/W</td><td>R/W</td><td>W</td><td>W</td><td>R/W</td><td>R/W</td><td></td><td>R</td></tr></table>	DE1	DE0	DWE1	DWE0	DIE1	DIE0	—	DME	0	0	1	1	0	0	1	0	R/W	R/W	W	W	R/W	R/W		R
DE1	DE0	DWE1	DWE0	DIE1	DIE0	—	DME																					
0	0	1	1	0	0	1	0																					
R/W	R/W	W	W	R/W	R/W		R																					
DMAモードレジスタ : DMODE		3 1	ビット リセット時 R/W	<table><tr><td>—</td><td>—</td><td>DM1</td><td>DM0</td><td>SM1</td><td>SM0</td><td>MMOD</td><td>—</td></tr><tr><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td></tr><tr><td></td><td></td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td>R/W</td><td></td></tr></table>	—	—	DM1	DM0	SM1	SM0	MMOD	—	1	1	0	0	0	0	0	1			R/W	R/W	R/W	R/W	R/W	
—	—	DM1	DM0	SM1	SM0	MMOD	—																					
1	1	0	0	0	0	0	1																					
		R/W	R/W	R/W	R/W	R/W																						

レジスタ	ニーモニック	アドレス	備 考									
DMA/WAITコントロールレジスタ :DCNTL	3 2	ビット	DM1,0		アドレス	SM1,0		アドレス				
			0 0	M	DAR+1	0 0	M	SAR+1				
			0 1	M	-1	0 1	M	-1				
			1 0	M	固定	1 0	M	固定				
			1 1	I/O	固定	1 1	I/O	固定				
			MMOD		モ ー ド							
			0	サイクルスチールモード								
			1	バーストモード								
			リセット時		MWI1	MWI0	IWI1	IWI0	DMS1	DMS0	DIM1	DIM0
			R/W		1	1	1	1	0	0	0	0
					R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
			MWI1,0		ウェイトステート		IWI1,0		ウェイトステート			
			0 0	0	0 0	0						
			0 1	1	0 1	2						
			1 0	2	1 0	3						
1 1	3	1 1	4									

レジスタ	ニーモニック	アドレス	備 考											
ILベクタレジスタ :IL		3 3	DMSi	DIM1,0	アドレス増減									
			1	エッジ入力	0 0	M→I/O	MAR+1 IAR固定							
			0	レベル入力	0 1	M→I/O	MAR-1 IAR固定							
					1 0	I/O→M	IAR固定 MAR+1							
					1 1	I/O→M	IAR固定 MAR-1							
			ビット	IL7	IL6	IL5	—	—	—	—	—			
			リセット時	0	0	0	0	0	0	0	0			
			R/W	R/W	R/W	R/W								
			INT/TRAPコントロールレジスタ :ITC		3 4	ビット	TRAP	UFO	—	—	—	ITE2	ITE1	ITE0
						リセット時	0	0	1	1	1	0	0	1
R/W	R/W	R							R/W	R/W	R/W			
ビット	REFE	REFW				—	—	—	—	CYC1	CYC0			
リセット時	1	1				1	1	1	1	0	0			
R/W	R/W	R/W								R/W	R/W			

レジスタ	ニーモニック	アドレス	備 考							
			CYC1,0	リフレッシュサイクル間隔						
			0 0	10ステート						
			0 1	20						
			1 0	40						
			1 1	80						
MMUコンベースレジスタ : CBR	3 8	ビット リセット時 R/W	—	CB <sub>6</sub>	CB <sub>5</sub>	CB <sub>4</sub>	CB <sub>3</sub>	CB <sub>2</sub>	CB <sub>1</sub>	CB <sub>0</sub>
			0	0	0	0	0	0	0	0
			R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
MMUバンクベースレジスタ : BBR	3 9	ビット リセット時 R/W	—	BB <sub>6</sub>	BB <sub>5</sub>	BB <sub>4</sub>	BB <sub>3</sub>	BB <sub>2</sub>	BB <sub>1</sub>	BB <sub>0</sub>
			0	0	0	0	0	0	0	0
			R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
MMUコン／バンクエリアレジスタ : CBAR	3 A	ビット リセット時 R/W	CA <sub>3</sub>	CA <sub>2</sub>	CA <sub>1</sub>	CA <sub>0</sub>	BA <sub>3</sub>	BA <sub>2</sub>	BA <sub>1</sub>	BA <sub>0</sub>
			1	1	1	1	0	0	0	0
			R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
I/Oコントロールレジスタ : ICR	3 F	ビット リセット時 R/W	IOA <sub>7</sub>	IOA <sub>6</sub>	IOSTP	—	—	—	—	—
			0	0	0	1	1	1	1	1
			R/W	R/W	R/W					

# 株式会社 日立製作所

電子事業本部  
電子部品営業本部

〒100 東京都千代田区丸の内一丁目5番1号(新丸ビル) 電話 東京 (03) 212-1111(大代)

栃木電子部品営業所

西那須野 (02873) 6-3312

厚木電子部品営業所 厚 木 (0462) 22-0763

多摩電子部品営業所

立 川 (0425) 27-0410(代)

新潟電子部品事務所 新 潟 (0252) 41-8161(代)

高崎電子部品営業所

高 崎 (0273) 25-2161

横浜電子部品事務所 横 浜 (045) 871-3929

松本電子部品営業所

松 本 (0263) 36-6632

沼津電子部品事務所 沼 津 (0559) 32-4962

茨城電子部品営業所  
(特 販 第 2 部)

勝 田 (0292) 74-4011

関西支店 大 阪 (06) 261-1111(大代)

東北支店 仙 台 (0222) 23-0121(大代)

京滋営業所 京 都 (075) 341-1771(代)

金沢営業所 金 沢 (0762) 63-2351(代)

九州支店 福 岡 (092) 741-5831(代)

中国支店 広 島 (082) 223-4111(代)

中部支店 名古屋 (052) 251-3111(大代)

四国支店 高 松 (0878) 31-2111(代)

北海道支店 札 幌 (011) 261-3131(大代)

マイコンについての総合情報センター

日立マイコンセンター "GAIN" (03) 253-1405(代)